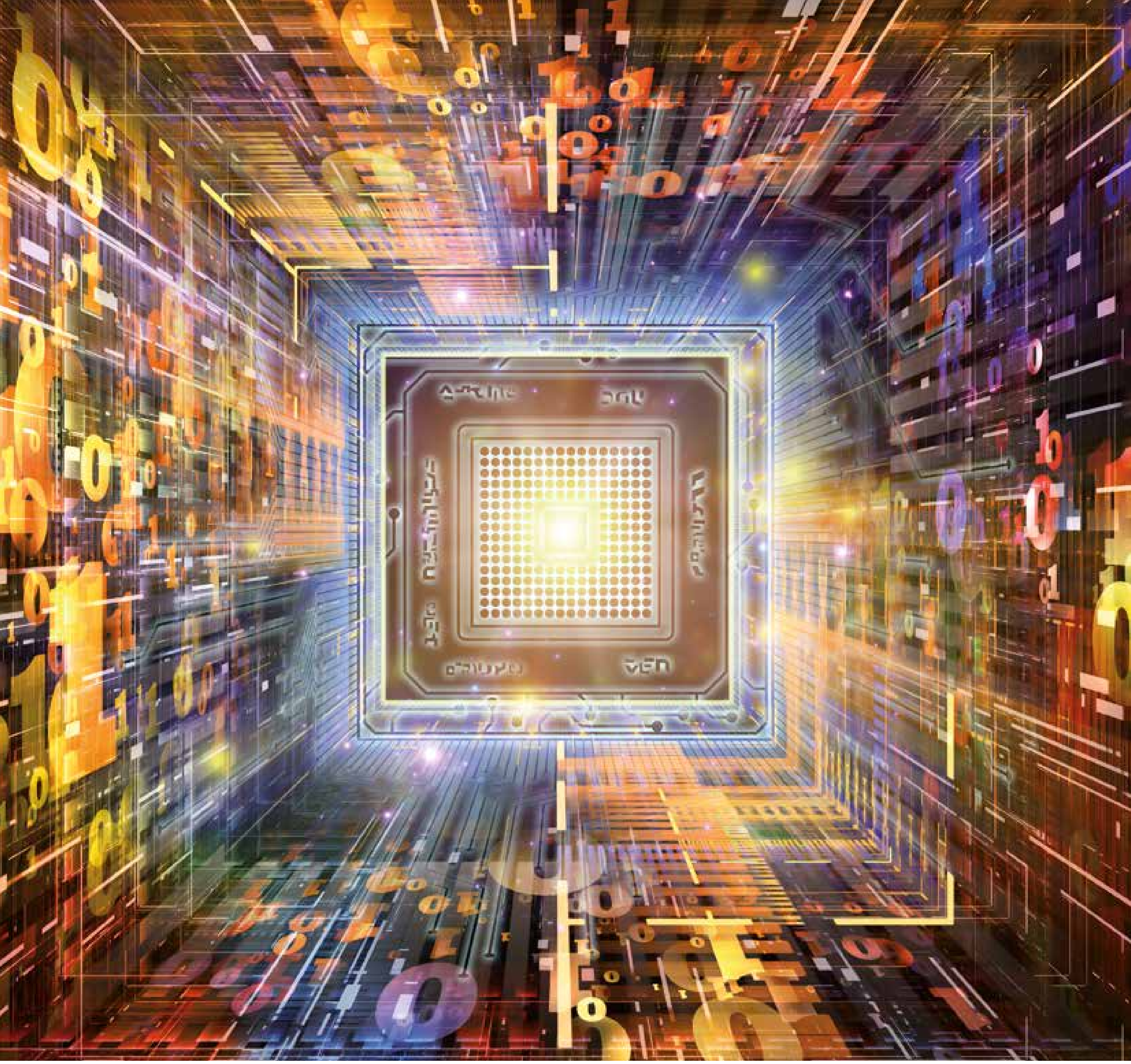


White Paper

Edge intelligence

Executive summary

The cloud is dead – long live the cloud!

Driven by the internet of things (IoT), a new computing model – edge-cloud computing – is currently evolving, which involves extending data processing to the edge of a network in addition to computing in a cloud or a central data centre. Edge-cloud computing models operate both on premise and in public and private clouds, including via devices, base stations, edge servers, micro data centres and networks.

Expansion of the IoT and digital transformation will generate entirely new businesses and markets, with both vendors and customers creating new demands on computing and networking infrastructures across all industries (automotive, aerospace, life safety, medical, entertainment and manufacturing, to name just a few).

Edge intelligence (EI) is edge computing with machine learning (ML) and advanced networking capabilities. This means that several information technology (IT) and operational technology (OT) industries are moving closer towards the edge of the network so that aspects such as real-time networks, security capabilities to ensure cybersecurity, self-learning solutions and personalized/customized connectivity can be addressed.

Container technology and ML are the leading technological responses to the demands of these new businesses and markets. Bringing these technologies to the edge, will fulfil the promise of EI.

Likewise, fifth generation wireless technology (5G) is bringing together IT and telecommunications, e.g. by enabling data centres at the edge of networks as well as the possibility of implementing industry-specific networks enabled by virtualization and software-defined networking principles in a single environment. Most of the anticipated 5G applications are driven by the IoT.

Based on a thorough review of such developments, this White Paper offers the following conclusions:

- Containerization will be important for EI, but no specific Standards exist in this area, although many open source initiatives have been developed, such as Docker and the Open Container Initiative (OCI).
- Common data models for edge computing node (ECN) communication are essential to the success of EI.
- Micro data centres will become more important, for a number of reasons, including the ability to provide low latency and to process large volumes of data, thereby avoiding their transportation to the cloud.
- 5G networks will provide data centres at the edge and the possibility to implement industry-specific networks enabled by virtualization and software-defined networking principles.
- The best user interface is no user interface. As IoT makes manual data input largely obsolete and ML and artificial intelligence (AI) take over decision-making, this becomes a reality.

This White Paper formulates recommendations based on a review of use cases versus existing technology and Standards. Additionally, this White Paper suggests that all recommendations be demonstrated and supported by the use of a testbed in collaboration with the Industrial Internet Consortium (IIC) to complement Standards with open source implementations.

Specifically, the present White Paper makes the following recommendations to industry:

- Prepare for disruption on business and commercial models.
- Utilize 5G Standards to facilitate edge computing and EI solutions.
- Include micro data centres in EI solution architecture, ideally employing containerization.
- Agree on a common approach to orchestration and lifecycle management to avoid market fragmentation.
- Agree on a common approach to ML (tools, model implementation) to avoid market fragmentation.

It is also recommended that the IEC take a larger role in promoting the development of the software component of electrotechnical systems. The IEC is in a unique position to drive EI forward.

Accordingly, this White Paper also outlines specific recommendations for the IEC Standardization management Board (SMB) in the areas of:

- Credibility and (decentralized) trust
- Self-organization, self-configuration and self-discovery
- Implementation of algorithms for ML

There already exist many standardization and consortium activities related to edge computing. This situation provides challenges with regard to optimizing edge computing standardization, but also opportunities to create a more positive standardization ecosystem that will support the needs of governments, industry and users.

Finally, such an ecosystem should be one of collaboration across the spectrum of standards development organizations (SDOs) and consortia. Specifically, this White Paper recommends that the IEC collaborate with IIC to complement Standards with open source implementations and testbeds. These include covering horizontal, vertical and specialty Standards, as well as EI Standards.

So how will the current technology ecosystem be influenced by these new concepts and the emerging technology? This White Paper provides insights and recommendations on various aspects of EI.

Section 1 gives a brief description of this new technology by highlighting the benefits and opportunities, as well as the challenges and foreseen Standards, enabling the development of the true potential of EI. The section concludes with a rough classification of edge computing scenarios.

Section 2 elaborates further on the evolution of computing models, how the IoT disrupts today's cloud-based network architecture, and how that disruption calls for a new computational model "on top of the cloud". Moreover, Section 2.4 provides a high-level novel edge computing architecture for EI.

Having established a general understanding of what EI is, section 3 provides a deeper insight on the trend drivers and needs matching EI in different industries. Moreover, an overview of what constitutes state of the art regarding the evolution of hardware, software and architectures for communication networks is included in this section to capture technological synergies, e.g. virtualization and the 5G system, and emerging technologies such as ML and blockchains.

Section 4 briefly covers a set of use cases from the manufacturing, smart cities, smart building and life safety areas, highlighting their requirements, observed gaps, needed capabilities for overcoming such gaps and the standardization topics that can be pursued in this regard. A detailed description of the use cases can be found in Annex A.

In section 5, further details are provided concerning specific gaps in the various use cases. The section concludes with a framework for discussion focusing briefly on common gaps between the analyzed EI use cases.

In section 6 the needed capabilities for all the use cases are extracted with a more detailed presentation of how the use cases would benefit from employing these capabilities.

In section 7 the identified missing Standards are described and an analysis on common missing Standards regarding the use cases is given.

In section 8, as a result of the use case analysis, a fusion of some of the use cases are described by the means of a testbed, along with the possible extensions.

Section 9 concludes this White Paper by presenting recommendations to industry concerning the possible benefits to be gained from employing EI, a series of general recommendations to SDOs and, finally, recommendations targeted specifically at IEC members.

Acknowledgments

This White Paper has been prepared by the edge intelligence project team, in the IEC Market Strategy Board (MSB), with major contributions from the project partner, the Fraunhofer Institute for Open Communication Systems FOKUS, and the project leader, Huawei Technologies Co. Ltd. The project team met four times – November 2016 (Berlin, Germany), January 2017 (Macau), March 2017 (Berlin, Germany) and May 2017 (Palo Alto, US). The project team is directed by Mr. Xuemin Wang, Director of the Industry Standards Department, Huawei Technologies Co. Ltd, a MSB member. The project team is listed below:

Mrs. Andreea Ancuta Corici, Fraunhofer FOKUS, Project Partner Lead

Mr. Yun Chao Hu, Huawei, Project Manager

Prof. Dr. Thomas Magedanz, Fraunhofer FOKUS

Mr. Marc Emmelmann, Fraunhofer FOKUS

Dr. Andreas Vogel, SAP

Mr. Tetsushi Matsuda, Mitsubishi Electric

Mr. Michael Cronin, Johnson Controls Inc.

Mr. Victor Kueh, Huawei

Mr. Bin (Remy) Liu, Huawei

Mr. Yaling Zhou, Huawei

Mr. Meigang Liu, Huawei

Mr. Yang Shi, Huawei

Mr. Kunlun Lu, Huawei

Dr. Marius Corici, Fraunhofer FOKUS

Dr. Mikhail Smirnov, Fraunhofer FOKUS

Dr. Alexander Willner, Fraunhofer FOKUS

Mr. Christopher Stephan Kertess, Fraunhofer FOKUS

Mr. Peter Lanctot, IEC

Dr. Gilles Thonet, IEC

Table of contents

List of abbreviations

Technical and scientific terms

AAA	authentication, authorization and accounting
AD	automated driving
ADN	application dedicated node
AE	application entity
AF	application function
AI	artificial intelligence
ANDSF	access network discovery and selection function
API	application programming interface
AS	application server
ASN	application service node
BLE	Bluetooth low energy
BSS	business support system
CapEx	capital expenditure
CCI	co-channel interference
CDN	content delivery network
C-ITS	cooperative intelligent transportation system
CNC	computer numerical control
CNN	convolutional neural network
CP	control plane
CPU	central processing unit
CRUDN	create, retrieve, update, delete, notify
CSE	common services entity
CSF	common services function
DCS	distributed control system
DNN	deep neural network
DSRC	dedicated short-range communications
DTLS	datagram transport layer security
D2D	device-to-device communication

ECN	edge computing node
EI	edge intelligence
EMS	element management system
eNodeB	evolved node B
EPC	evolved packet core
ERP	enterprise resource planning
E/W	east/west
FCAPS	fault, configuration, accounting, performance and security
FDD	fault detection and diagnostics
GIS	geographic information system
GPS	global positioning system
GPU	graphics processing unit
HMM	hidden Markov model
HPSL	high pressure sodium lamp
HSS	home subscriber server
IACS	industrial automation and control system
IAM	identity and access control management
IIoT	industrial internet of things
IN	infrastructure node
IoT	internet of things
IPE	interworking proxy entity
IRC	intelligent and resilient control
ISG	industry specification group (ETSI)
ISM	industrial, scientific and medical
ISP	internet service provider
IT	information technology
JAR	Java archive
LED	light-emitting diode
L1/L2	level-1/level-2 (cache)
LTE	long-term evolution
LWM2M	lightweight machine-to-machine protocol

MANO	management and orchestration
MEC	multi-access edge computing
MES	manufacturing execution systems
MI	machine intelligence
ML	machine learning
MME	mobility management entity
MN	middle node
M2M	machine-to-machine
NB-IoT	narrowband IoT
NAS	non-access stratum
NC	numeric control
NF	network function
NFV	network function virtualization
NIDD	non-IP data delivery
NoDN	non-oneM2M device node
N/S	north/south
NSD	network service descriptor
NSE	network services entity
ODP	open distributed processing
O&M	operations and maintenance
OAM	operations, administration and maintenance
OpEx	operational expenditure
OS	operating system
OSS	operations support system
OT	operational technology
PC	personal computer
PCRF	policy and charging rules function
PDN-GW	packet data network (pdn) gateway (gw)
PKI	public key infrastructure
PLC	programmable logic controller
PNF	physical network function

PoP	point of presence
ProSe	proximity services
PUF	physical unclonable function
QoS	quality of service
RAM	random access memory
rkt	Rocket container technology
RNIS	radio network information services
RSU	roadside unit
SCEF	service capability exposure function
SDN	software-defined network
SDS	software-defined storage
SGW	serving gateway
SIM	subscriber identity module
SMS	short message service
SoC	system on a chip
SON	self-organizing network
telecom	telecommunication
TLS	transport layer security
TOF	traffic offload function
TSN	time-sensitive networking
UE	user equipment
UI	user interface
UICC	universal integrated circuit card
UNB	ultra narrow band
UP	user plane
UWB	ultra wide band
VLAN	virtual local area network
VNF	virtual network function
VoIP	voice over internet protocol
VPN	virtual private network
VRU	vulnerable road user

V2I	vehicle-to-infrastructure
V2X	vehicle-to-everything
WAVE	wireless access in vehicular environments
WLAN	wireless local area network
XML	extensible markup language

Organizations, institutions, companies and protocols

3GPP	3rd Generation Partnership Project
5G PPP	5G Infrastructure Public Private Partnership
AMQP	advanced message queuing protocol
ANIMA	Autonomic Networking Integrated Model and Approach (IETF)
ARIB	Association of Radio Industries and Businesses (Japan)
ATIS	Alliance for Telecommunications Industry Solutions (US)
CCSA	China Communications Standards Association
CoAP	constrained application protocol
cuDNN	CUDA deep neural networks library (NVIDIA)
ETSI	European Telecommunications Standards Institute
HTTP	hypertext transfer protocol
IDC	International Data Corporation
IEEE	Institute of Electrical and Electronics Engineers
IEET	Institute of Ethics and Emerging Technologies
IETF	Internet Engineering Task Force
IIC	Industrial Internet Consortium
IP	internet protocol
Ipv4	internet protocol version 4
ISO	International Organization for Standardization
LPWAN	IPv6 over Low Power Wide-Area Networks (IETF working group)
MQTT	message queue telemetry transport protocol
MSB	Market Strategy Board (IEC)
NGMN	Next Generation Mobile Networks Alliance
OCI	Open Container Initiative
OCF	Open Connectivity Foundation

OMA	Open Mobile Alliance
OMG	Open Management Group
oneM2M	Standards for M2M and the internet of things
OPC	Open Platform Communications (Foundation)
OPC-UA	Open Platform Communications Unified Architecture
OSGi	Open Service Gateway initiative
OSI	open systems interconnection model
ROLL	Routing Over Low power and Lossy networks (IETF working group)
SMB	Standardization management Board (IEC)
SOAP	simple object access protocol
TCG	Trusted Computing Group
TCP	transmission control protocol
TIA	Telecommunications Industry Association (US)
TSDSI	Telecommunications Standards Development Society (India)
TTA	Telecommunications Technology Association (Korea)
TTC	Telecommunication Technology Committee (Japan)
UDP	user datagram protocol
WIA-PA	wireless networks for industrial automation process automation

Glossary

artificial intelligence
AI

a machine mimics cognitive functions that humans associate with other human minds, such as pattern matching, learning and problem solving.

access network discovery and selection function
ANDSF

3GPP core network component for 1) assisting user equipment (UE) to discover access networks and 2) providing the UE with recommendations regarding usage of the connection to these networks.

application programming interface
API

is constituted of clearly defined methods of communication between various software components.

convolutional neural network
CNN

is a class of deep, feed-forward artificial neural network that have successfully been applied to analyzing visual imagery (computer vision) and speech recognition.

deep neural networks
DNN

are feedforward artificial neural networks in which data flows from the input layer to the output layer without looping back.

datagram transport layer security
DTLS

a communications protocol that provides security for datagram-based applications.

edge computing node
ECN

a machine deployed close to the end device of an infrastructure, having the role to take local decisions based on sensed or received information and policies or algorithms from the core servers. The decisions can be applied (physically) locally or communicated to the core servers for aggregation and final decision.

edge intelligence
EI

edge computing with machine learning and advanced networking capabilities.

evolved node B
eNodeB

for LTE access network, it is the component that is connected to the mobile phone network that communicates directly wirelessly with mobile handsets (UEs).

evolved packet core
EPC

the first 3GPP core network having an all-IP architecture, no circuit switch components.

hidden Markov model
HMM

a statistical Markov model in which the system being modelled is assumed to be a Markov process with unobserved (i.e. hidden) states. The models are applied in reinforcement learning and temporal pattern recognition such as speech, handwriting, gesture recognition and bioinformatics.

industry specification group (ETSI)
ISG

ETSI groups that operate alongside traditional standards-making committees in a specific technology area. They are designed to be quick and easy to set up, providing an effective alternative to the creation of industry fora. [source: ETSI]

level-1/level-2 (cache)
L1/L2

level 1 and level 2 of central processing unit (CPU) cache. They are smaller, faster memory, closer to a processor core, which store copies of the data from frequently used main memory locations.

multi-access edge computing
MEC

ETSI specification that offers application developers and content providers cloud-computing capabilities and an IT service environment at the edge of the network. [source: ETSI]

machine learning
ML

for a specific field, machines are producing a model of the process they are observing or manipulating so that decisions based on observations can be made. The machines can be trained in order to improve the model.

network function virtualization
NFV

is a network architecture concept that uses the technologies of IT virtualization to virtualize types of network functions into building blocks that may connect create communication networks.

non-IP data delivery
NIDD

encapsulating UE related IP-data in signalling messages so that data transfer can be realized without the UE acquiring and IP address.

network service descriptor
NSD

in ETSI NFV MANO specification, each Network component/service is associated with a description, for the orchestrator to know how to manage the deployment of the service.

open distributed processing
ODP

uses UML for open data processing system specifications. It was introduced by the ISO/IEC 19793:2015 Standard.

operations and maintenance
O&M

general term that covers procedures to be used for deployment and maintenance of an infrastructure.

packet data network (pdn) gateway (gw)
PDN-GW

central component of 3GPP evolved packet core in charge of allocating an IP connection to the UE, enforcing QoS rules for the UE traffic and acting as mobility anchor when the UE handovers.

public key infrastructure
PKI

introduces a framework handling verification of certificates issued by certificate authorities for enabling services using secure electronic transfer, like e-commerce.

rocket container technology
rkt

is a new container runtime, designed for composability, security, and speed. [CoreOS.com]

software-defined networking
SDN

a technology to introduce software and APIs in routing components that can be provisioned with rules from components aware of the current state of the network in terms of attached users, available routing infrastructure.

time-sensitive networking
TSN

is a set of Standards for mechanisms for the time-sensitive transmission of data over Ethernet networks. The Standards are under development by the time-sensitive networking task group of the IEEE 802.1 working group.

vehicle-to-everything
V2X

safety and energy efficiency tailored communication for passing of information between a vehicle to any entity that may affect the vehicle. It is a vehicular communication system that incorporates other more specific types of communication as vehicle-to-infrastructure (V2I), vehicle-to-vehicle (V2V), vehicle-to-pedestrian (V2P), vehicle-to-device (V2D) and vehicle-to-grid (V2G).

Section 1

Introduction

The cloud is transforming, take a look at the edge

Historically, network architectures and computing models have oscillated between the use of shared and central resources and exclusive and local compute power. As of today, available massive distributed deployments of sensors and intelligent devices known as the internet of things (IoT) are confronted with the currently dominating cloud computing model emphasizing centralized shared resources. This model challenges the increasing use of mobile applications and use cases utilizing local resources and information gained from them.

In contrast, content delivery, IoT, and emerging information technology (IT) and operational technology (OT) applications are imposing increasingly stringent requirements on latency and bandwidth, demanding further optimization of network transport and data processing at or close to end devices. The limitations inherent in current commercial cloud approaches, which cannot be overcome via hyperscale cloud technologies [1], have motivated intensive research for innovative solutions to address issues such as distance and location of servers, latency and jitter, security and personal data privacy, proximity and location awareness of applications, and enhanced mobility support.

As such, a discrepancy exists between the state-of-the-art architecture, which – in a very generalized way – consists of data sources, data storage and data processing, and the content delivery network connecting such architecture, which is hindering the targeting of new use cases and markets such as industrial automation, robotics, e-healthcare or virtual reality (VR). All of these applications require the obtaining and processing of large amounts of information within an extremely short period of time, which increasingly translates into growing requirements for upcoming network architectures, notably the capability to transfer within milliseconds terabytes of (raw) data from their sources. It is obvious that in the near future, networks will not be capable of fulfilling those requirements. Even in the dawn of 5G networks, which will dramatically reduce delays in communication and increase available bandwidth, advances are mostly being achieved via geographically localized optimizations in the network.

Edge intelligence is born

The concept of edge intelligence (EI) introduces a paradigm shift with regard to acquiring, storing, and processing data: the data processing is placed at the edge between the data source (e.g. a sensor) and the IoT core and storage services located in the cloud. As such, the literal definition of edge and intelligence specified in Figure 1-1 is adopted: the ability to acquire and apply knowledge and skills is shifted towards the outside of an area, here the core communication network or the cloud.

EI allows bringing data (pre-)processing and decision-making closer to the data source, which reduces delays in communication. In addition, such (pre-)processing makes it possible to accumulate and condense data before forwarding it to IoT core services in the cloud or storing it, which perfectly matches the capacities offered by the upcoming fifth generation wireless technology (5G) networks providing localized throughput and delay enhancements. Edge computing [2]

Edge

noun

1. the outside limit of an object, area or surface
2. the sharpened side of the blade of a cutting implement or weapon

verb

1. provide with a border or edge
2. move or cause to move gradually or furtively in a particular direction

Intelligence

noun

1. the ability to acquire and apply knowledge and skills
2. the collection of information of military or political value

NOTE The term "edge intelligence" reflects a combination of the two concepts defined here.

Figure 1-1 | Dictionary definitions of "edge" and "intelligence"

makes computing and storage resources available in close proximity to mobile devices or sensors, complementing centralized cloud nodes and thus allowing for analytics and information generation close to the origin and consumption of data. Supplementary resources may even reside on end devices that might not be continuously connected to the backbone network. Additionally, EI allows future applications to depend on context awareness capabilities for mutual detection and proximity services, (near) real-time responsiveness for a tactile internet, data analytics at the edge and/or end device and device-to-device communication capabilities.

As processors, microcontrollers, and connectivity are embedded into a plethora of new devices, the application of EI in smart appliances, wearables, industrial machines, automotive driver assistance systems, smart buildings and the like continues to increase. For the purpose of enabling and realizing the true value of IoT, the trend toward adopting EI, which again pushes processing for data-intensive applications away from the cloud to the edge of the network, continues to expand.

1.1 Scope of this White Paper

The goal of this White Paper is to provide forward thinking for the next decade by exploring the benefits and opportunities of EI for vertical use cases, the technology drivers. The White Paper analyzes specific gaps posed by given use cases and their requirements and proposes new general capabilities.

In particular, this White Paper develops a vision of vertical EI having as its starting point today's use cases and projecting toward envisioned future use cases. The White Paper derives a natural synthesis from current trends in the areas of cloud computing, mobile networking, IoT and other domains that require increasingly low delays in communication and decision-making, e.g. smart manufacturing, video analysis for security and safety, automotive engineering, intelligent city furniture, and VR. IoT, with its myriad sensors and intelligent devices will trigger a Cambrian explosion of new services and applications, most of which are hardly imaginable today. These technologies will affect all facets of life and will cause disruptive transformations in all vertical industrial sectors. A major enabler will be the next generation of mobile networks offering support in ultra-low latency, high bandwidth and dependability. However advanced services for a tactile internet will demand fast data analysis, and cloud computing services will need to move close to the end device. The realization of these capacities will lead to a further fusion of previously separated fields and will require close cooperation among the various actors. These technologies will create value and new opportunities for involved stakeholders that extend to the vertical industries, but which will demand innovative types of cooperation based on new business models. Here standardization will constitute a key element in opening up former walled gardens and in realizing the vision presented.

1.1.1 Benefits and opportunities

By introducing intelligence at the edge computing nodes (ECNs), systems can:

- take decisions more quickly and efficiently by placing machine learning (ML) algorithms on the edge devices and reducing the frequency of contact with cloud servers, thus steadily reducing the effect of the roundtrip delay on decision-making;
- reach decisions according to local identity management and access control policies specific to the running applications, securing the data close to its source and following local regulations;
- lower communication costs by reducing communication over public wide area networks, using caching or local algorithms to pre-process the data so that only decisions or alarms can be forwarded to the cloud servers, rather than raw data;
- load-balance the user, application or network requests based on changes in the edge or core infrastructure, adapting to temporary failures or maintenance procedures;
- take decisions based on the alarms or pre-processed information exchange between the edge devices, i.e. east/west (E/W) communication between two peers on the edge.

These opportunities have resulted from the technological evolution of manufactured devices, software paradigms and core networks such as 5G. Coupled with these changes are new concepts, algorithms and Standards that have emerged around ML, software networks, mobile edge computing, analytics and identity and access control, for example blockchains.

1.1.2 Common challenges of edge computing scenarios

The compromise between required data volume and available bandwidth, the need for intermittent connectivity and the requirement for immediate responses as anticipated by edge computing scenarios characterize a wide variety of specific use cases, such as connected city lighting, smart elevators, life safety and security challenges and factory productivity improvement.

1.1.2.1 Needed Standards

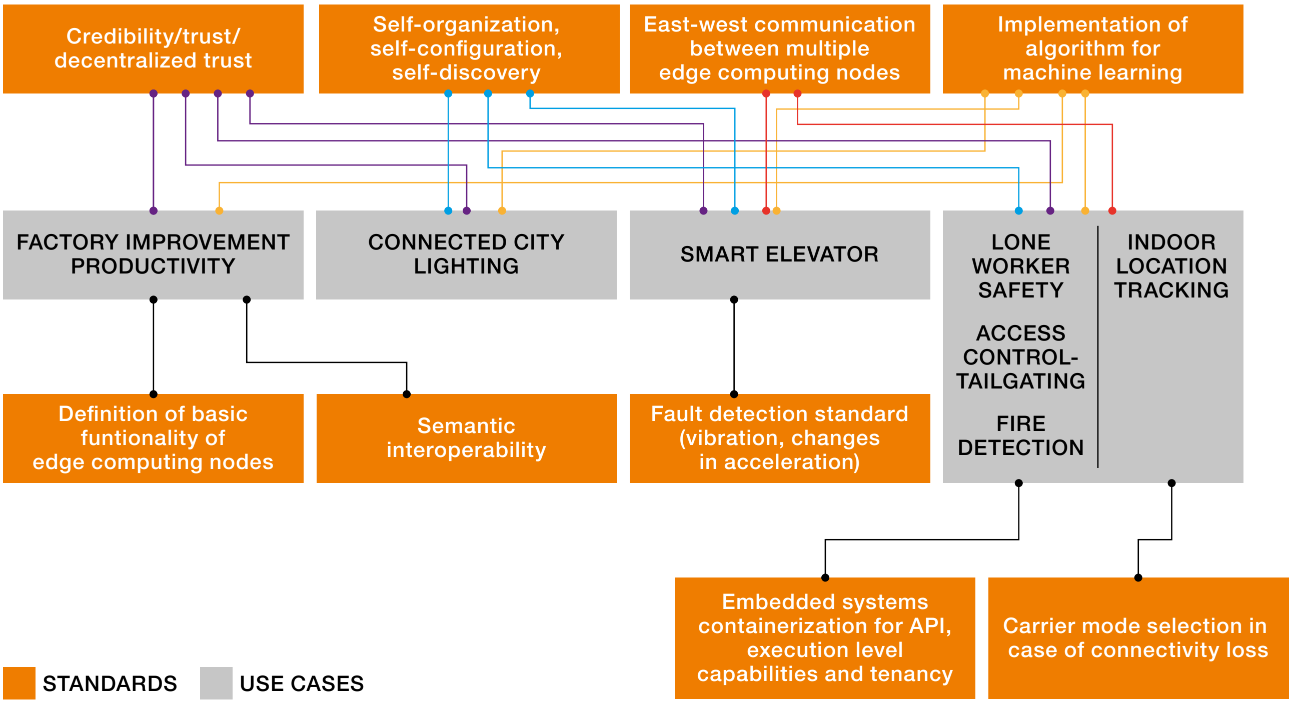

Credibility/trust/ decentralized trust
Self-organization, self-configuration, self-discovery
East-west communication between multiple edge computing nodes
Implementation of algorithm for machine learning
FACTORY IMPROVEMENT PRODUCTIVITY
CONNECTED CITY LIGHTING
SMART ELEVATOR
LONE WORKER SAFETY
ACCESS CONTROL-TAILGATING
FIRE DETECTION
INDOOR LOCATION TRACKING
Definition of basic funtionality of edge computing nodes
Semantic interoperability
Fault detection standard (vibration, changes in acceleration)
Embedded systems containerization for API, execution level capabilities and tenancy
Carrier mode selection in case of connectivity loss
STANDARDS
USE CASES

Figure 1-2 | Standards needed for EI

As shown in Figure 1-2, all of such use cases share a number of technical challenges and requirements that need to be considered when adapting EI:

- Credibility and (decentralized) trust
- Self-organization, self-configuration, and self-discovery
- E/W communication between multiple ECNs
- Implementation of algorithms for ML
- Definition of basic functionality of ECNs
- Semantic interoperability
- Fault detection Standards
- Embedded system containerization for application programming interface (API), and execution level capability and tenancy
- Carrier mode selection for avoiding connectivity loss

The Standards needed to address these challenges have been identified from an analysis of the use cases contained in this White Paper. Further details on these required Standards can be found in section 7.

1.1.2.2 Edge computing scenarios

1.1.2.2.1 Data volume versus available bandwidth

Devices and sensors can produce more data than is economically feasible to transmit to the cloud. To address this problem, analytical algorithms can be applied at the edge to process the incoming sensor data and only send higher level events to the core.

For example, tens or hundreds of cameras produce video streams at 60 frames per second. Even with compression, the transmission of video streams can be very costly. A video analysis service could be deployed at the edge that identifies people, objects (e.g. vehicles), and their properties (e.g. license plates and x,y coordinates). Only this higher level information would then be sent to the core.

The video content would be stored locally at the edge for a certain duration and could be accessed by a human operator for further analysis as needed.

IoT solutions are often cost sensitive, and communication costs specifically represent a significant portion of ongoing expenses. Low bandwidth wide area protocol solutions, such as LoRA, Sigfox and others, can reduce the communication cost. But these solutions come with the unwanted consequences associated with low bandwidth, such as reduced performance.

Thus, communication costs can be addressed more effectively by using analytical algorithms to process the incoming sensor data and only send alerts (another form of higher level events) to the core. This also enables confidential or privacy related data to be kept near the data source so that the disclosure of data can be limited.

1.1.2.2.2 Intermittent connectivity

When devices and sensors are in locations with only intermittent connectivity, they need local data processing and decision-making in order to keep operating.

For example, off-shore oil rigs and container ships use satellite connectivity which can be easily interrupted. Similarly, cars and trucks using cellular data connections can drift in and out of coverage as they move around.

IoT edge computing can provide data buffering as well as rules or predictive algorithms to allow for autonomous operation.

1.1.2.2.3 Immediate response

Decisions based on sensor data must often be made in real-time, if no time is available for a roundtrip to the core. For example, self-driving cars need to make decisions autonomously. Network latency would likely create severe safety issues.

Similar needs exist in process and discrete manufacturing industries where many parameters can influence the quality of the product. For example, sensor data are compared to a recipe derived through a multi-variant analysis. The golden batch, as it is known in process manufacturing, is created based on this analysis, with adjustments to temperature, pressure, humidity, etc. made in real-time.

Section 2

The evolution of computing models towards edge computing

2.1 Shared and central resources versus exclusive and local computation

The computing models of the last seven decades are now oscillating between the use of shared and central resources or exclusive and local compute power. Key factors in deciding the direction of the curve are advances in computing and communication. Cheap and powerful compute power pushes towards the use of local resources. Cheap and fast communication technologies enable the use of shared, central resources.

Models which relied heavily on shared resources included mainframes operated in batch mode or which were controlled by text-only terminals. At the opposite end of the spectrum, the standalone personal computers (PCs) of the 1980s were powered by affordable compute power.

Networked PCs and client-server models created a more balanced computing model between the two extremes enabled by high-speed local area networks.

The pendulum swung back towards central resources with the early web model enabled by cheap and fast wide area networking, compared to the level of data being transferred.

Today's dominating cloud computing model still emphasizes centralized shared resources, but mobile apps and JavaScript-heavy web applications often make good use of local resources.

Shared and centralized resources are highly efficient, as they maximize the utilization of compute resources and provide elasticity. Given their central location, typically in data centres, they can be more easily secured, and their lifecycle management is less complex than in distributed systems. However, they need highly available communication channels of sufficient bandwidth and speed to reach end users, which may incur significant cost.

Exclusive and local resources can work in isolation but the compute power, memory and storage are limited and may be insufficient for certain tasks.

As such devices are often under end user control, securing and managing them, as well as the lifecycle of their applications, becomes more complex.

The cloud computing model seems to have found a happy compromise in the distributed computing spectrum, balancing the pros and cons between exclusive/local versus shared/central, but this solution will not last.

2.2 IoT disrupts the cloud

The IoT disrupts the cloud compute model by introducing new usage scenarios resulting in the following key requirements:

- Real-time: often decisions need to be made within tens of milliseconds. Today's communication infrastructure and the laws of physics require local decision-making, as a roundtrip to the cloud would take an excessive amount of time.
- Connectivity: today's mobile networks are often spotty and cannot guarantee

connectivity to the cloud. Hence decision-making must occur locally.

- Data volume: the amount of data generated by sensors can be huge, for example hundreds of high-resolution cameras creating video streams at 30 frames per second, which could clog wide-area communication channels.
- Context: the business context needed for interpreting IoT data for decision-making is typically held in centralized enterprise systems.

The disruption of the cloud model is not the displacement of the cloud but rather its extension to the edge.

2.3 Characteristics of the new computing model

The cloud will continue to exist. For example, certain functions are best performed in the cloud, such as the training of predictive analytics algorithms, as typically only the cloud holds the necessary data in its entirety.

Devices will have compute and storage capabilities, for instance high-end security cameras can store and analyze video on the device.

Edge computing will provide compute power and storage in the space between the device and the cloud. Edge compute devices include IoT gateways, routers, and micro data centres in mobile network base stations, on the shop floor and in vehicles, among other places.

The new model will be a fully distributed computing model. It will support a wide range of interaction and communication paradigms, including the following:

- Autonomous, local decision-making based on incoming IoT data and cached enterprise information
- Peer-to-peer networking, for example security cameras communicating amongst themselves about an object within their scope
- Edge networking, for example platoon driving, i.e. vehicles self-organizing into groups which travel together, orchestrated and controlled by a micro data centre in the base station of a mobile network
- Distributed queries across data that is stored in devices, in the cloud and anywhere in between
- Distributed data management, for example data aging: which data to store, where and for how long
- Self-learning algorithms that learn and execute on the edge, or learn in the cloud and execute on the edge, or learn and execute in the cloud
- Isolation, involving devices which are disconnected for a long time, operating on minimal energy consumption to maximize lifespan

Through the introduction of intelligence at the edge nodes, systems can:

- take decisions more quickly and efficiently, as the roundtrip delay in contacting the cloud is removed;
- reach decisions according to local identity management and access control policies, securing the data close to its source;
- reduce communication costs by limiting communication over public wide area networks.

The opportunities come from technology evolutions in manufactured devices and 5G networks, along with concepts, algorithms and Standards in software-defined networking, mobile edge computing, analytics and device and data ownership.

2.4 Blueprint of edge computing intelligence

2.4.1 Definition and high level architecture

This White Paper defines EI as the infrastructure nodes that span from devices, corporate networks and public or dedicated networks up to the cloud deployments of the service, see Figure 2-1. All these nodes share the capabilities of processing data and applying algorithms to it, be it aggregation, caching, decision-making or routing. As these nodes evolve, they are also becoming capable of self-adaptation to data value, rate and access control and signal requests, so that the adjacent nodes also adapt. Today's most predominant concept for such aggregation of devices is the IoT, which connects smartphones, tablets and almost anything with a sensor on it – cars, machines in production plants, jet engines, oil drills, wearable devices, and more. These "things" collect and exchange data, interact with business systems and create significant business outcomes.

This means that several IT and OT technologies can be placed so close to the edge of the network that aspects such as real-time networks, security capabilities to ensure cybersecurity, self-learning solutions and personalized/customized connectivity can be addressed. This radical transformation from the cloud to the edge will support trillions of sensors and billions of systems and will treat data in motion differently from data at rest.

2.4.2 Application areas

2.4.2.1 IoT

From a network architecture perspective, the core of such an IoT solution is typically a central IT system, bearing the name of IoT core server, in charge of storing, processing and analyzing IoT data, see Figure 2-2. Much of this IoT data often can be located in the cloud, away from the core.

IoT endpoints (i.e. devices with sensors and/or actuators) frequently do not have the communication capabilities to transmit all their sensor data in a secure, reliable and cost-efficient manner to the core. The most common obstacles to such transmission include the following:

- Sensors may only support low energy protocols to conserve battery power.
- Mobile devices leveraging cellular communication lack coverage in certain locations.
- Mobile communication links are often bandwidth-constrained or expensive.
- Wide area connections can introduce too much latency for real-time decision-making.

Furthermore, certain local systems, for example self-driving vehicles, must autonomously make decisions in real-time and cannot wait for instructions sent from the cloud.

EI can address these challenges. An IoT gateway is an example of an ECN. It connects to devices that are located away from the core (often referred to as devices "at the edge") via communication protocols such as low-energy Bluetooth or ZigBee. At the same time, it also connects to the core directly

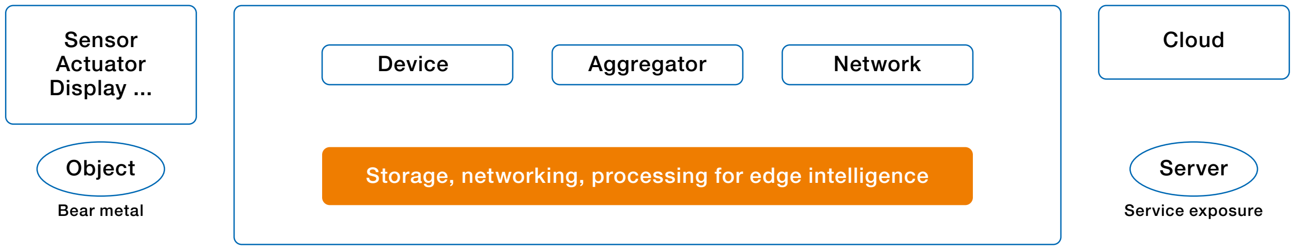


Figure 2-1 | Nodes capable of EI

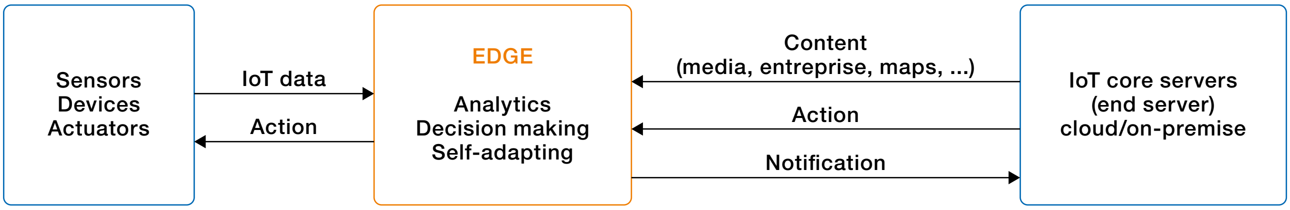


Figure 2-2 | IoT edge computing

using high-speed internet. Additionally, gateways provide security and lifecycle management at the edge, such that the edge is a sustainable and manageable compute unit. The hardware used for such gateways ranges from high-powered, rack-mounted servers to smaller devices with embedded processors, and anything in between.

IoT edge computing refers to the capability of processing, storing, and analyzing sensor data as well as performing decision-making at ECNs.

The role of the ECN is:

- to retrieve or pull IoT data from endpoints and content of a varied nature (e.g. media, enterprise bound, maps) from the IoT core servers, in order to be able to undertake data and networking analytics, take decisions on current information and self-adapt the local knowledge;
- based on the decision taken, to trigger an action towards the endpoints (e.g. actuate or change the threshold) or even send notifications towards the IoT core servers, e.g. request resources in terms of core computing, networking quality of service or dispatch of rescue forces, in case of fire or other dangerous situations.

Analysts, for example the International Data Corporation (IDC), indicate that 40% of IoT-created data will be subject to IoT edge computing, and that this ratio of edge-to-core data and processing is growing annually [3].

2.4.2.2 Content delivery networks

A content delivery network (CDN) is a distributed network of servers bringing content to end users. The goal of a CDN is to optimize content delivery with high availability and performance, while minimizing required bandwidth in the backbone and saving transportation costs. Instead of using a centralized server at a single location, content is delivered from servers situated near to the endpoints. With CDNs the traditional client-server model is split into two communication flows: one between end user and proxy media server and the other from the media server to the central server.

The advantages of CDNs include reduction of latency, limiting the impact of server and network failures and minimizing wide area transportation costs. A CDN also strengthens security. By being highly distributed, it can absorb the effects of less-sophisticated malicious attacks. The deployment of a CDN plays an essential role in the business strategy of content providers, leading to an improvement in the quality of experience of the customers. Enhancing user satisfaction represents a key factor for high conversion rates in online business, i.e. the number of website visitors actually performing desired actions such as purchase, subscription or ad-clicks.

CDN technology especially supports the delivery of large media files and streaming content, but other sites with heavy traffic that serve a large widely geographically distributed user community, e.g. social media or e-commerce in general, also benefit from CDNs.

Today CDNs serve a large fraction of the internet content. Such content may consist of web objects (text, graphics and scripts), downloadable objects (media files, software, documents), applications (e-commerce, portals), traffic from social networks and especially on-demand streaming media and live streaming media. A number of major companies specialize in the provisioning of CDN services, but support of CDN has also become part of the portfolio of global cloud services providers, internet service providers (ISPs) and network operators.

CDNs are derived from technologies for website acceleration, including server farms and intelligent caching. The CDN market started to develop in the late 1990s triggered by higher demand for audio and video streaming and growing volumes of content. With the further development of the technologies, additional factors such as cloud computing, energy awareness and user demand for more interactivity came into focus. Flash crowd phenomena observed in the context of events such as the 9/11 terrorist attacks created awareness concerning the importance of CDN solutions. The need for CDN services generated initiatives aimed at developing Standards for delivering broadband content and streaming rich media content (video, audio and associated data) over the internet.

The recent evolution of CDNs has been strongly driven by the continuing trend toward mobile end devices combined with a user expectation of receiving performance at least equal to conventional fixed or stationary devices. For creating a personalized interactive user experience, dynamic content generation needs to be supported with individually created suggestions and offers, without compromising download times and page rendering. As CDNs become increasingly sophisticated, the integration of multiple CDNs from different providers is often required.

CDNs form a major use case for increased EI. Media content is becoming even more localized, real-time and bandwidth-intensive. Hence more intelligence at the edge is needed to address these challenges.

2.4.2.3 Tactile internet

The capability to transmit touch in perceived real-time, which is enabled by suitable robotics and haptics equipment at the edges together with an unprecedented communications network, is often referred to as the “tactile internet” [4]. Thus, tactile internet stands for near real-time human-machine interaction, including cases in which the human is mobile. The use cases and opportunities enabled by the tactile internet are numerous and the performance requirements of networks are highly demanding.

The latency requirements for the tactile internet are very challenging, with a round trip delay of 1 ms or less typically required. 4G mobile networks can offer about a 25 ms latency under ideal conditions, which is way off the 1 ms mark required. 5G promises to deliver ultra-low latency for a number of critical use cases, including industry automation, robotics, remote surgery, etc. Such latency can only be achieved by deploying new hardware in the air interface as well as through deployment of edge clouds. Figure 2-3 demonstrates the latency requirements for the tactile internet. Furthermore, such latency requirements dictate the maximum distance from the sensor to the mobile edge cloud, restricted by the speed of light.

As shown in Figure 2-3, the use of edge clouds is necessary to fulfil the latency requirements of the tactile internet. However, edge clouds are also required in order to provide storage and computation for tactile internet services. Scalability, security and reliability are highlighted as critical characteristics of such edge computing in order to serve use cases such as remote surgery and industry automation, which are

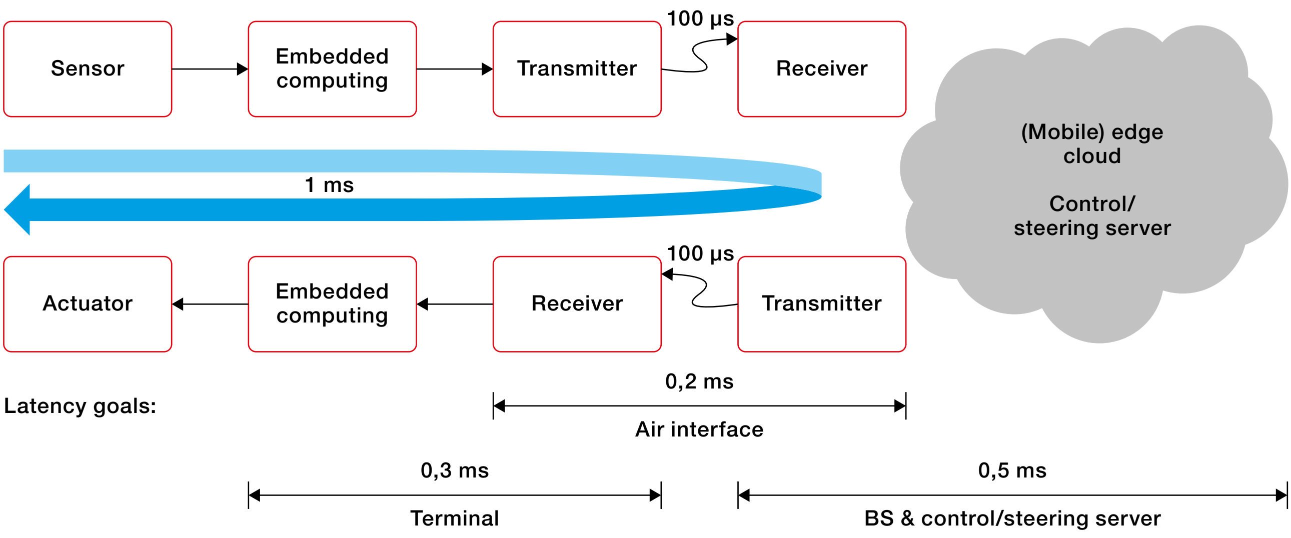


Figure 2-3 | Typical latency requirements for the tactile internet

described below. The deployment options of edge clouds determine a number of factors, including scalability and latency. Deploying small cloudlets at or very close to radio base stations can be useful to service-specific use cases, and deploying a variety of cloud technologies can ensure scalable and low power computing support for the tactile internet.

Augmenting edge computing with intelligence can further benefit the tactile internet. Latency and link speeds have certain physical limitations such as the speed of light. Deploying artificial intelligence (AI) and ML to enable support for a hybrid composition of machine and human actuation, mixing real tactile actuation with intelligence-based predictive actuation, can help to manage such limitations [4].

Examples of some uses of the tactile internet are described below.

- Industry automation: this is a steadily growing area for the tactile internet [5]. The sensitivity of control circuits, especially when controlling fast-moving endpoints such as robots, requires end-to-end latency of about 1 ms. Real-time feedback is imperative to ensure that a process or endpoint is operating correctly. Currently such systems are hard wired, using, for example, industrial Ethernet. In order to facilitate high flexibility, especially for smart manufacturing and Industry 4.0, wireless connectivity of machines is necessary. The nature of industrial automation, such as complex chemical processing plants or precision manufacturing, requires that connectivity remains highly available at all times and that communications between endpoints are secured. Cloud computing and storage to allocate resources on demand can provide the necessary scale and reliability to serve tactile internet services.
- Robotics: remote-controlled robots can be deployed to perform tasks in situations not accessible by humans, such as in a natural disaster or nuclear power plant maintenance. In such cases real-time, synchronous and visual-haptic feedback are necessary to ensure correct operation [4]. Reliable, low-latency communication links are required to ensure control reaction times are fast enough

for the robots and the objects with which they are interacting.

- Healthcare: tactile internet may facilitate remote surgery whereby a surgeon can control a robot to operate on a patient in a different location. High precision feedback in visual, audio and haptic form is crucial for such intricate actions. In the case of telemedicine or telesurgery, the need for high availability and privacy is extremely important.
- Virtual reality: enhancing the audio-visual aspect of VR with the ability to touch is an added benefit that can be facilitated by the tactile internet. Ensuring that the latency between feedback to various senses is low is important to avoid disorienting the user. Social, gaming and collaborative use cases can be envisaged which can benefit from a more immersive VR experience.

Education, gaming, autonomous vehicles, exoskeletons and even smart grids are some of the other use cases that could greatly benefit from the tactile internet [5].

Section 3

Trend drivers and state of the art for edge intelligence

This section introduces the anticipated EI technologies within telecommunication and industry vertical sectors in 2020 and beyond.

3.1 Industry needs

In this section, a short analysis of the needs of the following industries is presented: manufacturing, automotive, smart building/life safety, assets/utility management, smart grid, entertainment and transportation.

Some of the most stringent needs of the analyzed industries that can be solved through placing intelligence on the edge computing units include:

- Mobility: the automotive, transportation and manufacturing industries are the most demanding in terms of networking support (mobility and wireless broadband), requiring a high degree of mobility, especially in terms of handovers. At the same time, the quality of service and session handover management are critical aspects that can benefit from intelligence in the network components.
- Ultra low latency in decision-making: for the automotive, transportation, manufacturing and smart building industries, decisions on detection or actuation have to be taken within a delay of less than tens of milliseconds. For this, local intelligence residing on the edge computing units can help lower the delay and achieve the targeted response time.
- Autonomy: a chief requirement for the automotive, transportation, manufacturing, smart grid and smart building/life safety use cases is autonomy to cope with situations when the connection to the core server or service (e.g. power service) is no longer available, to prevent damage to persons, goods or infrastructure.
- Security: for the automotive, transportation, manufacturing, smart grid, smart building/life safety and asset management sectors, security is a feature that can never receive too much attention. Access control to physical or virtual resources (e.g. data) has to be ensured. Locally provisioned or learned policies and other mechanisms for authentication and authorization running on edge computing units are envisioned to enable fast adaptability of the systems.
- Local network bandwidth: for manufacturing, automotive, transportation, smart building/life safety and entertainment use cases, having a local bus bandwidth to enable local communication between the components is extremely valuable for being able to stream information in case of detection of accidents or other dangers, as well as for offering a competitive service (e.g. good video streaming for entertainment). Thus, intelligent routing (e.g. involving awareness of the context) on the edge computing unit can help to better respond to such requirements.
- WAN network bandwidth: the entertainment industry is the most demanding with regard to this need, for which the network intelligence on the edge computing units can play a pivotal role in obtaining a good quality of steaming, by allocating priority and introducing caching.
- Peer-to-peer communication: almost all of the industries make intensive use of peer-to-peer

communication, except for assets/utility management and smart grid, in which access control decisions are made either locally or by engaging the core service provider.

- Prioritization: in the smart grid, manufacturing and automotive sectors, a strong need exists for prioritizing communication based on the data carried, be it from the end device to the core server or vice versa. Edge computing units running intelligent software for detecting the type of traffic involved and the current data urgency can be put in place in order to enforce the prioritization of specific traffic or decisions.
- Self-organization, discovery: assets/utility management, entertainment and manufacturing to a certain degree require discovery of the capabilities of the devices and services and their role in the infrastructure, so that operations can be handed over from humans to intelligent software.
- AI/ML: in the fields of transportation, manufacturing, smart building/life safety and smart grid, as well as in the other industries, there is a concrete need for – and potential gain from – new algorithms that can adapt the communication and decisions or generate alarms about the system without human intervention, but rather with the help of adaptive software and the interaction of edge computing units.

A broad view of industry needs on different capabilities of the EI are summarized in Figure 3-1 using a heat map to suggest the importance of the needs for a specific industry. The value scale is between 0 and 100, with 0 value considered in case the feature is not needed and 100 value in case the feature is of utmost importance for the industry.

Need	Industry sector							
	Manufacturing	Automotive	Smart building/ life safety	Asset/ utility mgmt	Smart grid	Consumer IoT	Entertainement	Transportation
Mobility	55	98	10	50	10	55	80	97
Ultra low latency (<10ms)	95	100	85	5	5	15	15	95
Autonomy	95	100	100	7	100	50	45	100
Security	100	100	100	90	100	25	30	100
Local network bandwith	100	100	90	10	10	35	90	100
WAN network bandwith	35	30	55	15	10	55	90	45
Peer-to-peer communication	80	90	85	10	50	90	85	100
Prioritization	100	100	15	45	90	10	55	45
Self-organization discovery	60	50	20	95	40	65	90	60
Artificial intelligence/ machine learning	100	60	100	65	85	45	60	95

Figure 3-1 | High level needs for EI by industry sector

3.1.1 Buildings and life safety industry

Current edge solutions are based on low intelligence, low network intensity solutions. Such systems have the following common features.

- Mostly not networked, except for the largest systems
- Characterized by a low level of intelligence and configurability
- Operating in a custom hardware environment
- Marked by a low compute level, with no containerization

Examples of the types of systems currently deployed in the industry are shown in sections 3.1.1.1 to 3.1.1.3.

3.1.1.1 Data centre suppression – localized processing – low order

In the example shown in Figure 3-2 there is no significant computing capacity, and the level of networking is low. Systems of this type are limited in their evolution by the approvals required to update the equipment.

3.1.1.2 Security systems – localized processing – low and middle tier – cloud connectivity

In contrast, the one area where high levels of EI are deployed is the area of video surveillance, with some level of containerization beginning to be present, as can be seen in Figure 3-3.

Systems such as those shared by internet protocol (IP) camera networks are highly networked, with distributed metadata capabilities on the cameras themselves, for example object recognition, whereby the object specification is given to an application on the camera and the video stream being uploaded is annotated with that data, see Figure 3-4.

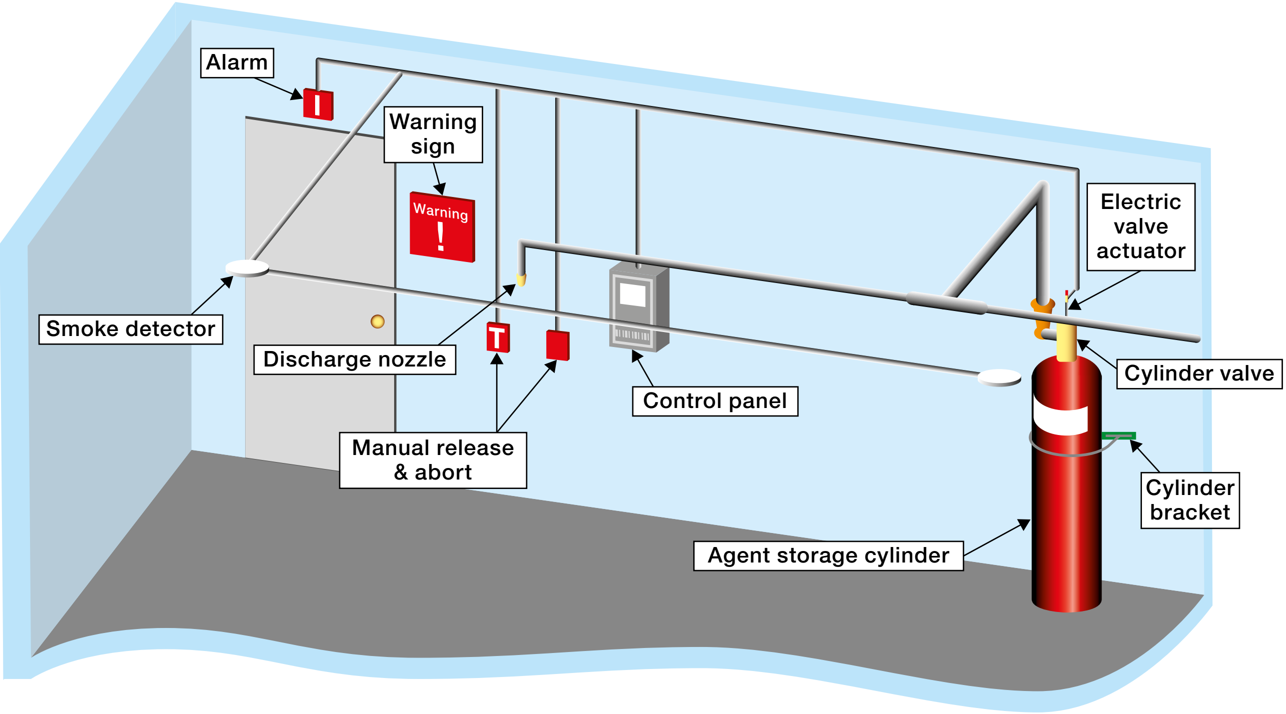


Figure 3-2 | Typical traditional suppression – data centre

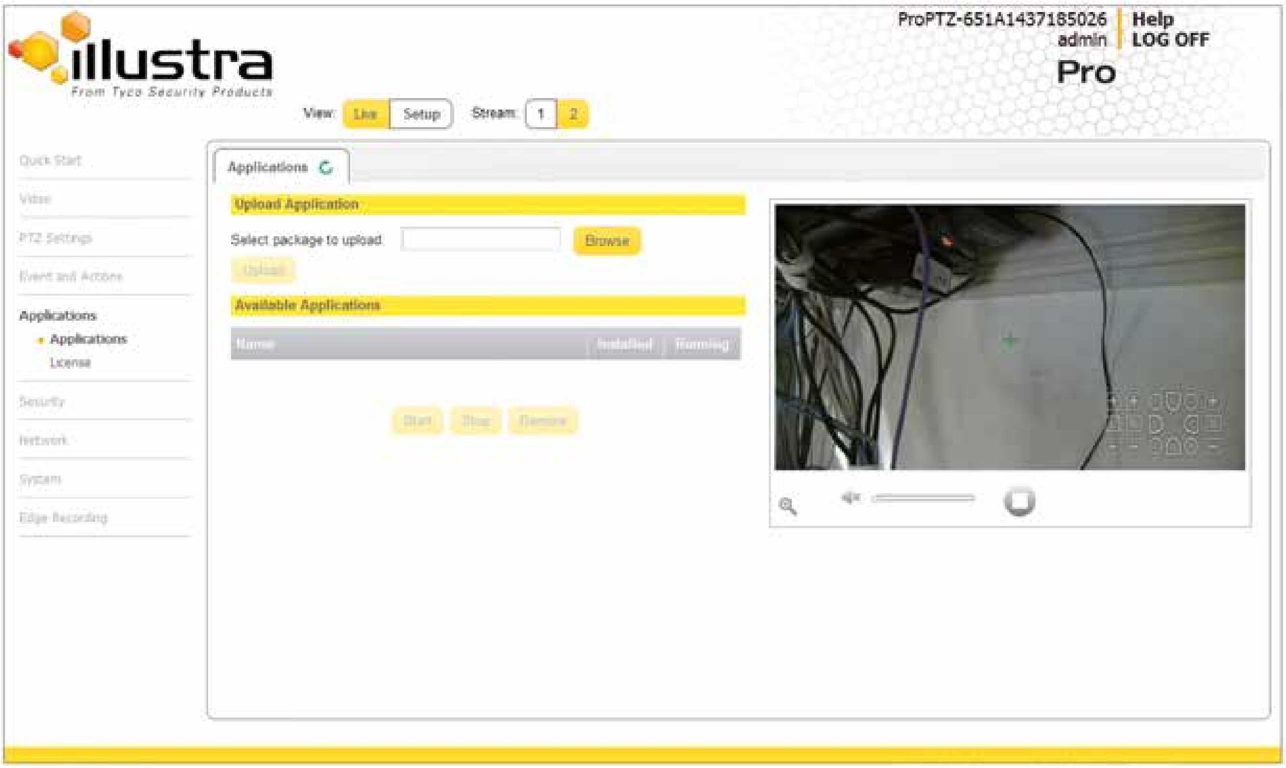


Figure 3-3 | Camera user interface (UI) showing modular capability

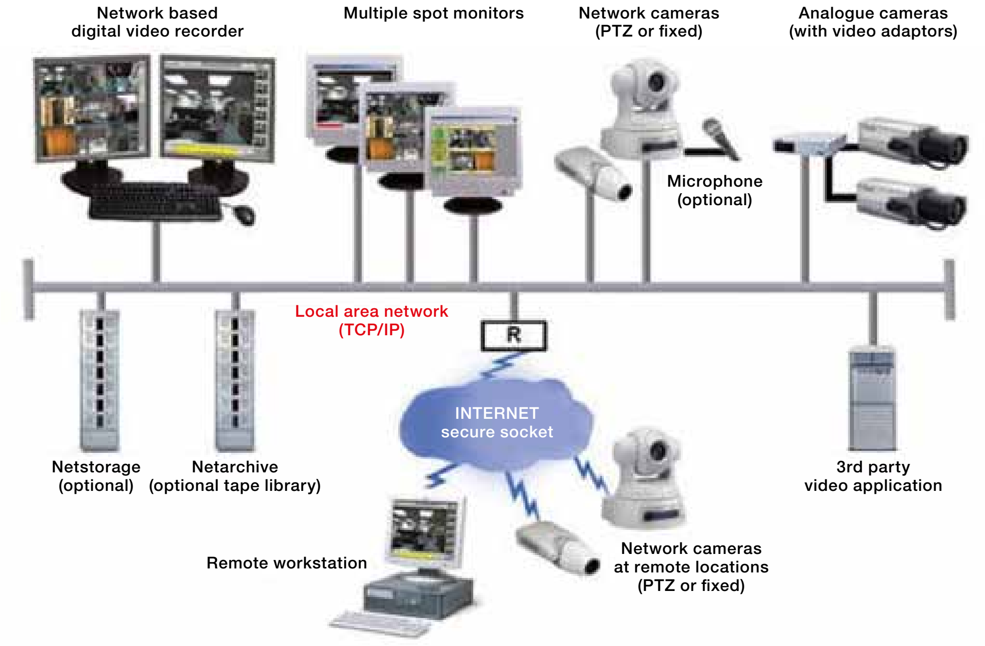


Figure 3-4 | Typical IP camera network

Such streams have metadata added, then get consolidated and uplinked to a local centralized console, as can be seen in Figure 3-5.

This console offers feature functionalities such as centralized pan and zoom, object tracking and others. Within the current state of the art, these systems depend on strong centralized control of rules and configuration. Further development will migrate more rules and intelligence down onto the camera nodes, allowing more decision-making to be made at camera levels.

The current state of the art would constitute a solution at this level, with live streaming capability to the cloud and cloud storage, but would never serve as the primary method of storage. An opportunity exists here for 5G to remove the in-premise element.

Figure 3-5 | Centralized control centre

3.1.1.3 Building management – high order – cloud analytics

With the advent of cloud computing and custom applications, users have come to expect solutions involving deep analytics that are not only scalable, but tailor made for their needs. Cloud-based technologies bring such solutions to the connected building, assembling data from equipment and building systems and presenting normalized, unified information that has been customized to user needs. Data from sensors, smart equipment, chiller plants and metres can be aggregated and analyzed alongside data that originates outside a facility, such as utility bills, and information that affects a facility's operations, such as weather forecasts or energy prices. This rich cache of data offers users a bigger picture of their building or chiller plant operations and efficiency. Cloud analytics enable remote monitoring of smart equipment, so experts in another location can spot a fault or trend that indicates a chiller is not running optimally. They can even diagnose the root cause of the issue and dispatch a service technician to make repairs before a real problem causes equipment downtime. Another solution might offer fault detection and diagnostics (FDD) on the equipment level, alerting the facility manager to the problems underlying a specific inefficiency, and in many cases catching abnormalities before the building automation system alarms are generated. Enterprise options allow users to look across their entire portfolio, accessing dashboards and workflow tools to gain clarity and insight. When information is aggregated across the portfolio and presented through a single interface, accessible anywhere, users have the tools they need to make intelligent decisions quickly and gain more control over their buildings.

Figure 3-6 shows a gateway controller normally used for translation between local sensor nodes and a cloud system.

Figure 3-6 | Network automation controller

3.1.2 Smart manufacturing

The industry needs identified in the IEC MSB White Paper *Factory of the future* [6] include the following specific needs related to smart manufacturing.

- Connectivity and interoperability: this designates the ability of a system to interact with other systems without application of special effort for integration. The dimensions to be considered are "vertical integration", which includes factory-internal integration, from sensors and actuators up to enterprise resource planning (ERP) systems, "horizontal integration", which includes the integration of production networks on the business level and "integration towards engineering and product/production lifecycle applications" in order to enable low-effort knowledge-sharing and synchronization between product and service development and manufacturing environments.
- Architecture for integrating existing systems: it is necessary to establish appropriate IT system architectures that support the stepwise implementation and extension of factory of the future systems, i.e. the modular roll-out of respective solutions. For the implementation of such an architecture, several needs that have to be considered are "device management and integration" and "persistence mechanisms".
- Security and safety: system boundaries are extended and the number of interfaces to remote systems increases. So do access points for potential threats from outside, which results in a need for appropriate IT security and safety measures. Moreover, system complexity increases with the increasing number of system components and the connections between them, which might cause unintended back coupling effects or the accidental overlooking of risks.

In addition to the above needs, the need for interoperability (portability) of application software across different hardware platforms will emerge with the proliferation of applications for smart manufacturing. Definitions of functionalities provided by platforms, and interfaces to access those functionalities, will be beneficial for meeting this need. Reliability of the system is also needed, so that manufacturing does not stop unintendedly.

Figure 3-7 shows the basic architecture of state of the art for smart manufacturing solutions (e.g. eF@ctory [7], PLAT.ONE [8]).

Sensors and actuators (e.g. robot, numeric control (NC)) are connected to programmable logic controllers (PLCs) or distributed control systems (DCSs), which communicate with them through a field network or field bus and control them in real-time. Some sensors (e.g. a video camera with a video analytics function) and actuators (e.g. an intelligent robot) have computing and/or storage capabilities. Usually only limited computing and/or storage capabilities are supported by PLCs and/or DCSs. ECNs (e.g. industrial PCs, gateways) are connected to PLCs/DCSs and sensors/actuators directly through a field network/bus or indirectly through an information network (e.g. Ethernet). ECNs usually have richer computing and/or storage capabilities and some edge computers may act as a gateway between OT and IT. An IT system can be implemented on premise or in the cloud.

These solutions provide connectivity and vertical integration in one way or another. The level of security, functionalities and interfaces to the functionalities provided by these solutions is different from solution to solution and is proprietary.

3.1.3 Automotive

The on-going evolution of IT and OT technology in the automotive industry embraces all facets of the changes affecting the sector: the reorganization of manufacturing processes in the wake of the industrial internet of things (IIoT), changes in

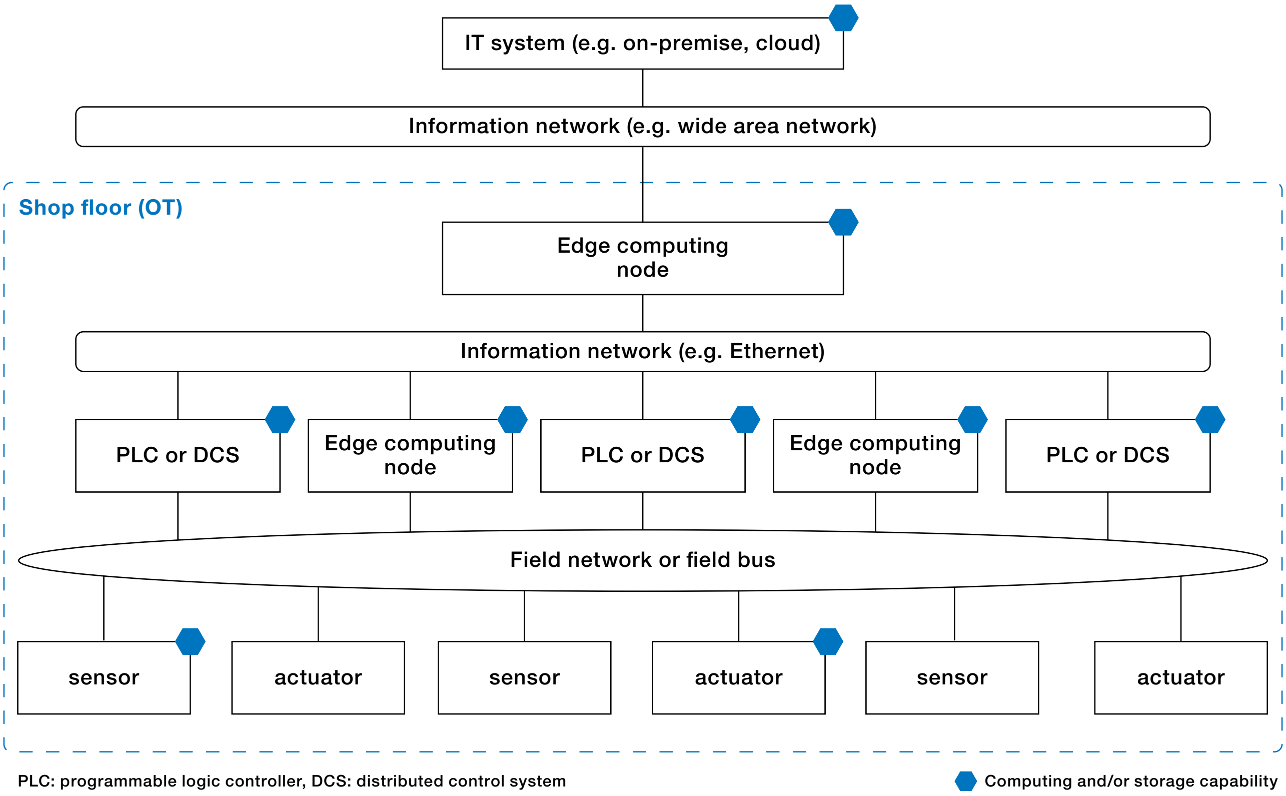

IT system (e.g. on-premise, cloud)
Information network (e.g. wide area network)
Shop floor (OT)
Edge computing node
Information network (e.g. Ethernet)
PLC or DCS
Edge computing node
PLC or DCS
Edge computing node
PLC or DCS
Field network or field bus
sensor
actuator
sensor
actuator
sensor
actuator
PLC: programmable logic controller, DCS: distributed control system
Computing and/or storage capability

Figure 3-7 | Basic architecture of state of the art for smart manufacturing solutions

consumer behaviour driven by e-mobility concepts and the demand for infotainment while on the road, challenges posed by global climate change and calls for renewable energy sources, and completely new approaches to logistics and mobility enabled by autonomous, self-driving vehicles.

All these trends and visions are bringing car manufacturers, IT vendors and the telecommunications industry into close collaboration. Within the 5G Infrastructure Public Private Partnership (5G PPP) initiated by the European Commission, representatives from the European automotive and telecom industries released in October 2015 their “5G Automotive Vision” on the next generation of connected and automated driving and new mobility services [9]. Public administrations and industry are setting out roadmaps for the deployment of cooperative intelligent transportation systems (C-ITS) for cooperative, connected and automated mobility and for the roll-out of uninterrupted 5G coverage of all urban areas and along the major transportation routes by 2025 [10] [11].

Vehicle-to-everything (V2X) technologies often apply short-range communications, in particular specially adapted wireless local area network (WLAN) Standards, to vehicular communications. Wireless access in vehicular environments (WAVE) is an approved amendment to the IEEE 802.11 WLAN Standard family [12]. However where appropriate this will be complemented by emerging 5G technologies.

The following use cases are specific to the automotive industry and bring together EI requirements:

- Automated driving: a major driver in the development of C-ITS technologies is the research on automated driving (AD) [13], involving autonomic vehicles (driverless or robotic cars) that are capable of navigating without the input of a human driver. These vehicles are capable of sensing their environment by making use of a variety of techniques such as radar, laser and global positioning system (GPS). Conditional automated driving is projected for 2020, while highly automated driving of driverless intelligent vehicles deployed in cites is foreseen by 2030.

- Cooperative intelligent transportation systems: use cases and their early deployment in field trials are central to the development of C-ITS technologies. Cooperative examples for automated driving, in which a vehicle may depend on information beyond the range of its sensors and may need coordinated decision-making include automated overtake (e.g. see-through vision, where video information from the front car is received and integrated), cooperative collision avoidance and high-density platooning.
- Road safety: use cases for road safety and traffic efficiency services profit from connected vehicles that periodically exchange status information (position, speed, acceleration, etc.) or environment information (e.g. traffic congestions, risks from weather conditions), thereby generating collective knowledge through collective perception. With vulnerable road user (VRU) discovery, pedestrians or cyclists can discover vehicles in proximity via their mobile devices and also announce themselves to such vehicles. Cameras or radar sensors at intersections can provide streaming information to approaching vehicles to supplement the information from on-board sensors.
- Remote sensing: within digitalization of transport and logistics, remote sensing and control functions profit from the connected vehicle. Health status information of components can be used in predictive analysis to detect impending faults. Remote processing performed on cloud servers can support limited local capabilities on the vehicle, e.g. for generating an augmented reality (AR) display on the windshield.
- In-car AR and media streaming: AR and real-time video streams can also enhance navigation systems by presenting real-time traffic conditions. Users will expect the same levels of connectivity in the vehicle as in the home or at the workplace. For infotainment services and a mobile workplace, high data rates and low latency connectivity will be needed on the vehicle. On the other hand it will become possible to use resources on the vehicle as nomadic nodes for improvement of mobile networks, e.g. utilizing unused communication and computing resources of parked vehicles as small cells to improve the capacity, data rate, energy efficiency and/or coverage of the mobile network.

All these use cases demand that the edge nodes be intelligent in the sense that the communication adapts to the local requirements for both machine-to-machine (M2M) and content world applications by providing low latency communication for using the application server (AS) deployed on the edge for functionalities such as local data aggregation or caching, see Figure 3-8. At the same time, the state of the communication has to dissolve (be distributed) on the edge network to make services ubiquitous.

Technical requirements for networking demand dedicated short-range communications (DSRC) providing one-way or two-way short-range to medium-range wireless communication channels [14]. Device-to-device (D2D) communication and long-term evolution (LTE) proximity services make it possible to bypass the infrastructure and transmit data directly among users, to offload traffic from the infrastructure and achieve a significantly lower delay. These features became available with

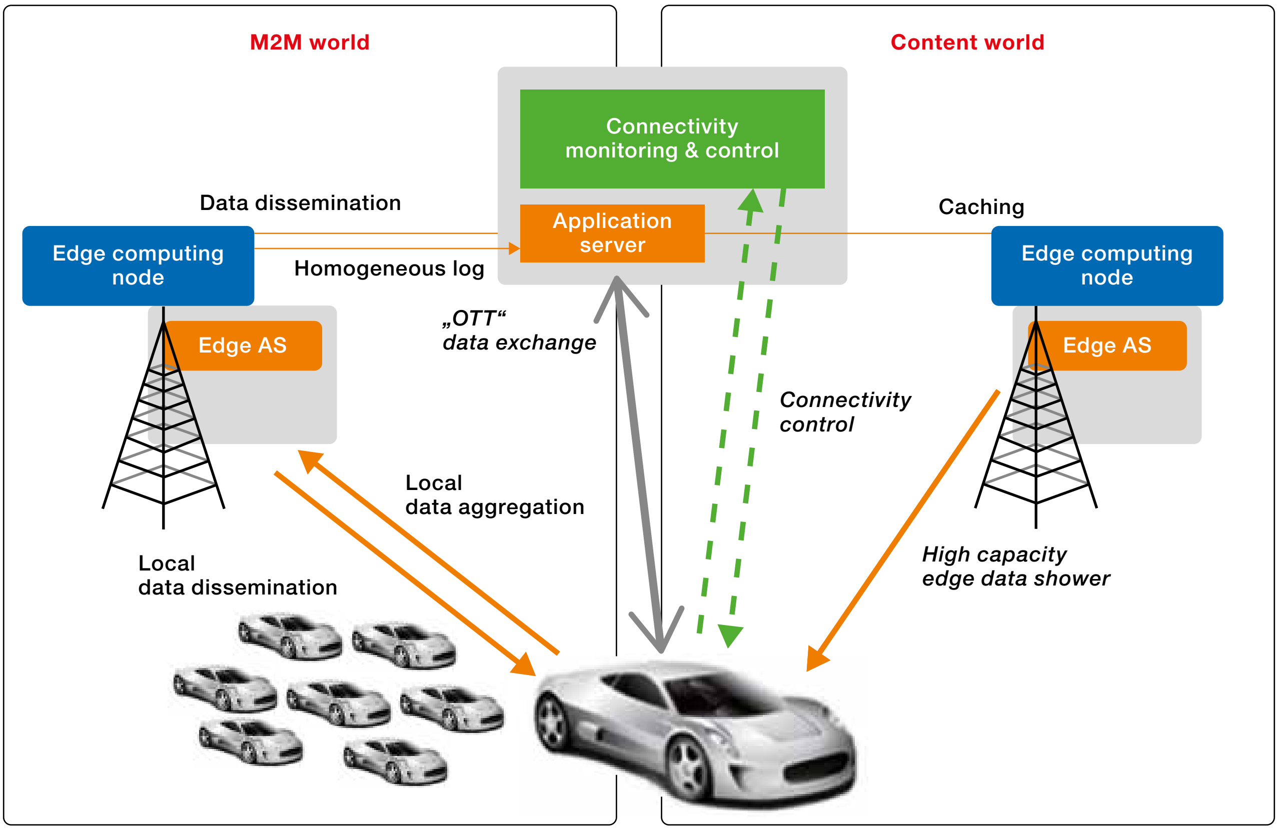


Figure 3-8 | Automotive EI for both M2M and content world

3rd Generation Partnership Project (3GPP), Release 12 (in July 2015), however further enhancements are currently under development to better fulfil the requirements of V2X communications. Cellular V2X communication based on 5G will support superior V2X capabilities and possibilities [15].

3.1.4 Information and communication technology

3.1.4.1 Dedicated localized networks

Dedicated networks have long been used for public safety and military networks. This is an example of the natural course of technology landing in domains where public interaction is taking place.

There is an increasing demand on the part of participants in public events to be able to consume personalized media while attending or participating remotely in sports, music or similar activities, e.g. taking virtual tours, accessing emergency maps, using AR or games to further increase the human experience.

This need is focused around broadcasting, VR or AR, compatible delays requiring low delay processing, secure communication including identity management and access control.

Other domains having similar requirements include:

- Industry 4.0, in which personnel can be helped or trained to use new machinery for assembly or repair [16]

- Safety applications, in which the communication has to be totally isolated and which provide low delay for the communication while at the same time ensuring the resilience of a network, either the same one or a different one, for the case of enabling the participant to return to safety [17]

EI can manifest by addressing:

- Localized access control or geofencing of data generated locally
- Analytics for real-time sensor data or other inputs to be processed for monitoring safety
- Local core networks to cope with the communication traffic and not disturb the national operator
- Support of personalized plug and play inputs in the multimedia streams

Technologies currently provide support for such networks, for example:

- Through device-to-device communication
- Proximity services (ProSe) in 3GPP defines a mechanism to generate groups of devices that are authenticated by the network identity management component, the home subscriber server (HSS) [18]
- Mobile edge computing is providing a valid hook for applications to run close to the end user, nearby the base station (see section 3.4.5.5)

3.1.5 Security domains

In this section an overview is given of information presented in the framework of standardization, research and specific projects with respect to security domains, especially for cases in which the role of the entity changes, joins or leaves another security domain.

A security domain is a collection of people, data, systems and devices that comply with the same security policy. A security domain contains components for identity and access control management [19].

For the past few years the effort of Standards was focused on considering interaction between two security domains from the human-to-service perspective, in which a user would like to access a remote service from security domain A and would use its own identity provider from security domain B to authenticate to the remote service.

Recently the IoT domain has introduced the case of a device running a service generating or consuming data. One owner per application is able to access the information that is stored on the device or is being forwarded to a remote service. However, as the IoT domain is becoming pervasive, with device manufacturers, consumers integrating devices in a system and end users needing to have access and ownership to the device simultaneously or not, an increasing need for flexibility is emerging for being able to change the owner of the device to guarantee the privacy of the data. The owner should then be able to reconfigure the IoT device to send data to its security domain without having the entire data streaming through the device manufacturer or the integrator security system. Identity and access control management (IAM) at the device level is also necessary for reconfiguration or actuation.

3.1.5.1 Dynamic identity and access control of the data or device owner

One of the key requirements for future communication from the IAM is flexibility of connectivity and service establishment. For this, changing the ownership of data and allowing the application to process sensitive data in the simplest and most secure manner is important for use cases such as automotive, digital life and energy, see Table 3-1.

Table 3-1 | Flexible IAM use cases

Use case	Actors	Required features
Automotive	Sensor manufacturer, car manufacturer, car owner, car repair shop, car second owner, car sharing driver	Allow access and right to manage car resources dynamically, e.g. for actuating (load/unload shipment); on-the-fly car sharing; session continuity
Digital life	Citizen, bank, online shop, transport company, governmental institution, service provider	Easy to use identity, identity wrapper over many identities, service discovery
Energy	Industrial energy provider, citizen as provider, citizen as consumer, smart home user, smart home manufacturer, smart home application developer, cloud provider	Dynamic coupling of services and devices to enable privacy

3.1.5.2 Supply chains case study

In order to enable fast time to market and privacy, services can be organized in dedicated virtualized networks including a service orchestrator to deploy new services [20].

An adapted definition of the security domain consists of the network deployed for a use case, together with the supported components stretching over multiple layers: core network, IoT communication management (including access control), IoT data collector and actuating engine. Two security domains that are related to a device can be deployed at different locations or they can also overlap, meaning that a device can be part of multiple security domains at the same time, see Figure 3-9.

A key requirement in order to reduce maintenance costs is a limited human intervention throughout the life cycle of a device, including during service session setup until the service session finishes.

Privacy and trust in sensitive areas such as automotive or smart delivery are of increasing concern to the possible client, be it a consumer or an industrial enterprise.

In order to enable new services or to make existing ones more flexible, a solution is needed to allow devices to use their local identity management intelligence to join new security domains dynamically and confidently, to adapt the communication automatically in order to establish the service sessions inside the new security domain, and at same time to preserve privacy [21] [22].

3.1.6 Blockchain

Blockchain technology allows for the creation of decentralized trust and reliance between two or more parties in the form of storing, moving and managing value, for example currencies, in such a way that the existence of huge governmental systems will be dispensable in the future [23]. The technology itself is a well-ordered distributed database that maintains a list of all transactions and which grows continuously over time [24].

3.1.6.1 Smart contracts concept

Smart contracts are suites of algorithms and protocols that value and control property by programmes running on behalf of persons [25].

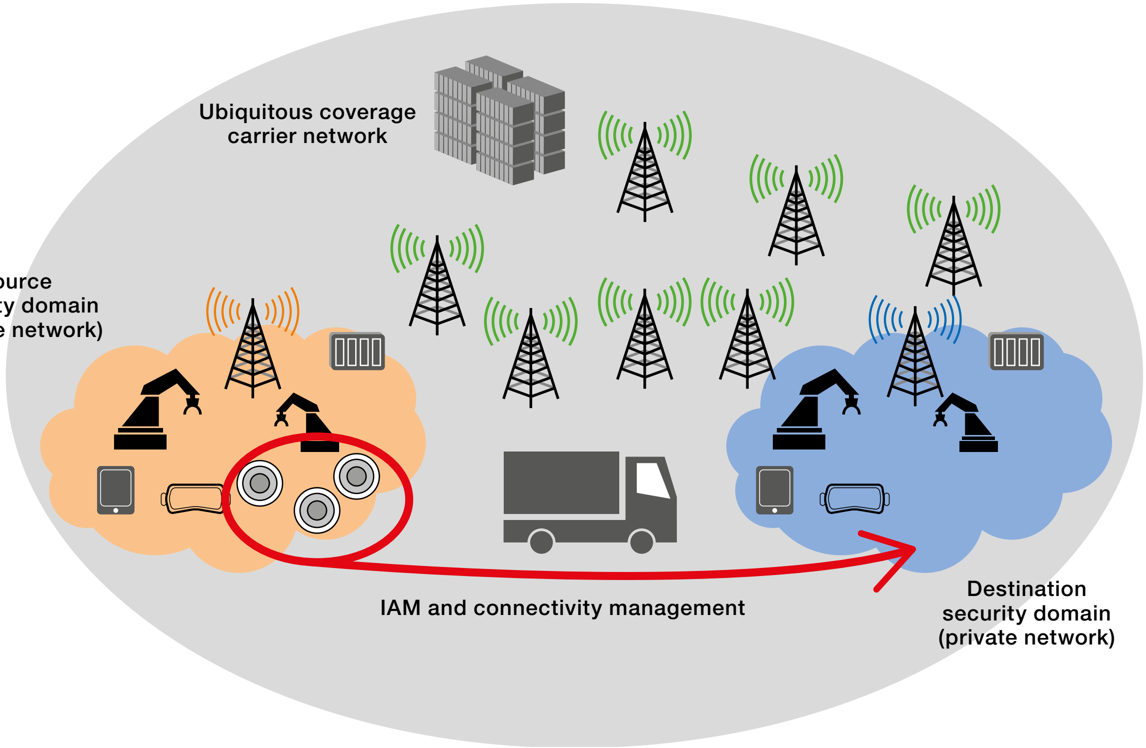

Ubiquitous coverage carrier network
Source security domain (private network)
IAM and connectivity management
Destination security domain (private network)

Figure 3-9 | Flexible security domains [source: IEEE]

A recent survey paper divided blockchain-inspired technologies into two categories: fully decentralized permission-less ledgers (e.g., bitcoin, Ethereum) and semi-centralized permissioned ledgers (e.g., Ripple) [26]. Smart contracts are by definition public, including in bitcoin. Private contracts are a more powerful variation that can handle private information, i.e. their state is not strictly public.

3.1.6.2 Technology insight

To understand how the blockchain technology works, it is important to look at the structure of a single block, the chaining/mining process of the blocks, enabled by its distributed network character, and the cryptography which enables the sending and receiving of bitcoins, the most popular implementation of blockchain.

- **Structure of a block**

To begin with, every single block consists of a body and a header. The header (80 bytes in size) holds the identification information used to prove the authenticity of both the block and its containing transactions. According to [27], the header consists of three main different sets of metadata information, see Table 3-2.

- The first metadata contains the previous block hash that links the current block to the previous block in the blockchain.
- The second piece of metadata describes the merkle root, a data structure used to summarize all the transactions that occurred in the block.
- The third set of metadata contains the difficulty, timestamp, and nonce, which all relate to the mining of new blocks.

The body (or content) contains a validated list of all digital transactions which occurred between the creation of a given block and its predecessor in the chain, using a specific data structure called a merkle tree or binary hash tree. This tree is made by recursively hashing pairs of nodes, using a specific SHA256-algorithm, until there is only one left called the merkle root, which is part of the header.

Considering an average transaction size of 250 bytes and 500 transactions in each block, the body is around 1 000 times larger than the header and therefore makes up the main size of a block.

- **Chain of blocks**

The chain of blocks contains a certain number of transactions. In other words, multiple transactions are grouped into one block. If the body of a block reaches its maximum size, the following transactions are placed in the next block. These blocks hang sequentially one behind the other and form a long, distinct chain. This well-ordered structure is achieved by the previous block hash, which every block inherits in its metadata so that it can systematically refer to the previous block in the chain. The first block in the chain is called the genesis block, and since it does not have a predecessor block, the genesis block is programmed, in contrast to all the following blocks, which are calculated. The height of a block is the number of all blocks in the chain between it and the genesis block. Consequently, the genesis block has a height of zero, since the zero block preceded it.

- **Decentralized technology**

Unlike traditional centralized databases, the blockchain is not located and maintained on a single computer (server) that belongs to a central authority such as a bank. Participants in the network (nodes) have a complete copy of the entire blockchain (starting from the genesis block) in their local storage by synchronizing their own copies of the chain with those of other users in the network via a wallet. If a new block is created in the network, certain participants use their computers to identify and verify each block in the chain, and in case the majority of the participants agree that the new block is valid, the new block is approved by the network and the blockchain grows in size and gets broadcasted to the whole network.

3.1.6.3 Technology adoption and evolution

The use of blockchain technology continues to grow rapidly, with many applications and systems being developed continuously by various parties, such as financial institutions, technology corporations and start-ups. To distinguish the degree of maturity and scope of these applications, Melanie Swan, Affiliate Scholar of the Institute of Ethics and Emerging Technologies (IEET) and principal of the MS Futures Group, introduced

Table 3-2 | Blockchain block metadata

Size	Field	Description
4 bytes	Version	A version number to track software/protocol upgrades
32 bytes	Previous block hash	A reference to the hash of the previous (parent) block
32 bytes	Merkle root	A hash of the root of the merkle tree of this block’s transactions
4 bytes	Timestamp	The approximate creation time of this block
4 bytes	Difficulty target	The proof-of-work algorithm difficulty target for this block
4 bytes	Nonce	A counter used for the proof-of-work algorithm

three categories: Blockchain 1.0, Blockchain 2.0 and Blockchain 3.0 [24].

3.1.6.3.1 Blockchain 1.0 – cryptocurrencies

Blockchain 1.0 entails the deployment of cryptocurrencies. The applications deal with the execution of payment activities through specific transaction-enabling software built "on top" of the blockchain, such as bitcoin, the most successful representor of Blockchain 1.0.

Bitcoin is a decentralized digital currency, introduced in 2008, that is recently gaining popularity due to the protocol of distributing smart contracts for reaching agreements with individuals. Transactions of smart contracts are exchanged in a mesh network in which peers are users of the bitcoin currency. The software running on behalf of the bitcoin users is charged with checking transactions. As timestamps cannot be used to validate the order of transactions and avoid double spending, the blockchain concept was introduced. A block contains multiple transactions and a pointer to a previous block, similar to a linked list. An algorithm to select/choose the next block of transactions is already defined, in order to reach an agreement over a highly distributed network to make double spending almost impossible. The algorithm selects the longest block branch. To generate a new crypto hash transaction, and thus a possible block, requires a lot of computing power. A small door remains open to malicious intent, in which a node having 50% of the computing power of the network has a 50% chance of winning and exercizing influence for another block branch to be used.

What is important from the identity management perspective is that a human or company can have multiple identities in the bitcoin network, as the element used as identity is a public key infrastructure (PKI) pair (private key/public key). The emitter, the one giving an amount of bitcoin currency (BTC), will use its private key to digitally sign the transaction. His public key is used by the network peers to check the transaction validity. The receiver of the BTC will use his/her private key to consume the BTC. The bitcoin network includes nodes that do not need any trust between them, as mathematics and digital signatures protect the correctness and integrity. One of the disadvantages is that the nodes are virtually anonymous and can lead to illegal operations, as governments cannot control or track the transactions.

3.1.6.3.2 Blockchain 2.0 – smart properties

In contrast to Blockchain 1.0, which focusses solely on cryptocurrencies, Blockchain 2.0 includes wider economic, market, legal and financial transactions. Such transactions include various values such as stocks or bonds, but also tangible and intangible assets such as cars and patents, identity information such as a driver's licence and public/private records such as birth certificates, signatures and licences, see Table 3-3. For any values of this kind it is important to have accurate records that identify the current owner and prove that he/she is the real owner of the value. Hence, the representation of these digital assets in the blockchain is needed, the so-called smart property, which is described as "the notion of encoding every asset to the blockchain with a unique identifier such that the asset can be tracked, controlled, and exchanged (bought or sold) on the blockchain" [24].

Due to Blockchain 2.0's generic concept, various problems that exist in today's (digital) world can be resolved.

3.1.6.3.3 IoT

Through the IoT, billions of technological devices such as sensors, near field communication (NFC) tags, iBeacons, Wi-Fi, Bluetooth, etc. can be linked. However, despite the huge market

Table 3-3 | Overview of transaction classes in Blockchain 2.0

Transaction classes	Examples
Financial transactions	Stock, bonds, private equity, crowdfunding, mutual funds, derivatives, annuities, pensions
Public records	Land and property titles, vehicle registrations, business licenses, marriage certificates, death certificates
Identity management	Driver's licenses, identity cards, passports, voter registrations
Private records	Loans, contracts, bets, signatures, wills, trusts, escrows
Attestation	Proof of insurance, proof of ownership, notarized documents
Physical assets	Home, hotel rooms, rental cars, automobile access
Intangible assets	Patents, trademarks, copyrights, reservations, domain names

potential of such devices, the IoT faces two major challenges [28]:

- First of all, the participating devices have to be correctly identified by the other devices in the IoT-network, which is becoming increasingly difficult considering the amount of different tasks, data formats and manufacturers of the individual devices.
- Secondly, after the correct identification, the data exchange must be trustworthy.

Blockchain solves these two problems via the combination of smart properties and smart contracts, since both are based on the fact that two or more parties do not need to know one another (each device can be registered directly by the manufacturer as an own entity in a universal blockchain, linked to product information) or even trust one another (due to the consensus mechanism). Consequently, it can be said that the blockchain is essential to unlocking the potential of the IoT" [23].

A very good example illustrating such contracts is the Autonomous Decentralized Peer-to-Peer Telemetry (ADEPT) project [29] for the IoT, which was jointly developed by IBM's Institute for Business Value and Samsung. IBM integrated a blockchain software in a Samsung washing machine that could operate completely autonomously in the consumables, energy and service market, without the need of human intervention. Based on the Ethereum blockchain that was selected because of its ability to implement smart contracts, the washing machine could order and pay for its own laundry detergent when the detergent was running low, enabled via smart contracts between the owner and the contracted service provider. Moreover, in the event of a system failure, the washing machine autonomously could order a craftsman and – depending on the warranty status in the machine's blockchain – could pay him accordingly. In addition, the washing machine was taught how to communicate with the local grid. A smart contract was concluded with the owner that allowed the community members to do a certain number of laundry wash cycles at the contract partner's washing machine, in return for a certain amount of energy that the washing machine needed to operate.

3.1.6.3.4 Blockchain 3.0

Blockchain 3.0 applications go beyond transactions in the areas of currency, finance and business and touch qualitatively on areas such as government, health, science, education, culture and the arts, but also social values such as liberty, equality, and empowerment. Moreover, Melanie

Swan also hints to the fact that these applications could lead to a higher magnitude of collaboration between individuals and corporations in the society and business world, but also between humans and intelligent machines. She even goes so far as to say that “perhaps all modes of human activity could be coordinated with blockchain technology to some degree, or at a minimum reinvented with blockchain concepts by requiring consensus to operate”.

3.1.6.4 EI evaluation

As a technology that is providing a decentralized framework for validating transactions, blockchain technology can be used by ECNs, although the advantages and disadvantages of the technology have to be evaluated when adopting it to a specific domain. Table 3-4 summarizes these aspects.

3.1.6.4.1 Advantages

- **Transparency and traceability**

The key strengths of an unpermissioned or permissionless blockchain architecture such as bitcoin is that its ledgers are open and accessible to everybody in the network. The bitcoin ledger, for example, shows a complete history of every single bitcoin transaction which has ever taken place in the blockchain and adds precise tracking information via the timestamp. This transparency becomes highly relevant in industry branches whose products are characterized by complex supply chains with different parties involved that must meet high quality standards, for example the food industry. Since all the important transaction data is stored in the distributed register and is never deleted, the auditors can trace the history and context of all transactions. This makes the blockchain a highly transparent and reliable system.

- **High availability**

If one or more server fails in the network (for technical reasons or because a participant ceases to operate), the existence of the ledger is not endangered, since there are still numerous registers that remain in operation due to the others. Moreover, the network becomes more robust when new users join and additional copies are made.

- **Privacy (pseudo-anonymity)**

When node A makes a transaction, there is no reference to either party’s real identity. This is because nodes interact with the network via a generated address (the public key), hence no identification is necessary to participate in the network, such as a credit card number, address, real name, etc. Blockchain systems are generally described as anonymous. However, since the flow

Table 3-4 | Advantages and disadvantages of blockchain technology

Advantages	Disadvantages
High transparency and traceability	Danger of “51% attacks”
High availability	High complexity
Pseudo-anonymity	Loss of private keys
Programmable	Misuse of pseudo-anonymity
No central governance	
No trusted third parties fees (high efficiency and lower costs)	

of transactions is traceable and therefore reveals the digital identity of a node, the term pseudo-anonymity appears to be more appropriate, since a public key serves as a pseudonym.

- **Programmable and additional information**

The block itself can include information that goes beyond the transaction details, such as ownership and personal records. The block is programmable, meaning that certain rules, such as if/when-clauses, can be applied that lead to a desired action.

- **No trusted third parties fees**

Over a period of many centuries trade has become incredibly complex, mainly through globalization. Trade is recorded in bookkeeping and this information is often isolated and closed to the public. For these reasons, trust is placed in third parties and middle brokers to facilitate and improve transactions, such as governments, banks, and accountants. These trusted third partiers apply heavy fees for exchanging the assets. The blockchain technology greatly reduces these fees by eliminating third party intermediaries and overhead costs. Moreover, current payments between financial institutions can take a long time, due to clearing and final settlements, particularly outside of working hours. Conversely, blockchain improves the efficiency of transactions by increasing the speed of payment between financial institutions, since transactions can be processed 24/7.

3.1.6.4.2 Disadvantages

- **Mining pools and “51% attacks”**

In bitcoin, mining pool participants add up their computational resources to increase the chance to mine bitcoins and therefore to share the rewards. However, this collective mining mechanism inherits the risk of the establishment of large industrialized mining pools, leading to a small amount of unique mining members which substantially threatens the decentralized character of the blockchain. Consequently, this increases the risk of a malicious mining pool attempting a “51% attack”. Although this scenario is highly unlikely, it could lead to reduced confidence in the network. A solution to this problem might be to employ signed transactions and a validating party that can be called upon to verify the authenticity of the transactions.

- **High complexity**

One of the main challenges facing blockchain is that there is a widespread lack of understanding about how this technology works in detail. In contrast to standardized internet protocols such as Ipv4 and applications such as hypertext transfer protocol (HTTP), the development of blockchain technologies is still in its very early stages. Although many companies understand the huge benefits that this highly technical and complex technology can offer, at the same time, this complexity acts as a barrier for individuals and businesses who want to use it. This lack of understanding also applies in cases where companies or individuals are exposed to cyber attacks. In the case of the bankruptcy filing of the Mt. Gox exchange, the company stated that “unspecified weaknesses” in the system led to the attack, indicating that full anticipation and understanding of the bitcoin technology itself, including potential attacks and countermeasures, is nearly impossible.

- **Loss of private keys**

Although the technology employs strong cryptographic protocols, its security might be compromised by service providers or users who fail to safeguard their private key on their computers. In case of a loss of the private key, the value of the data transferred to this key is lost forever. For bitcoin this leads to an unknown size of the actual monetary base of a cryptocurrency, which in turn can destabilize the economic dimension of bitcoin, caused by a high amount of apparently dead coins.

- **Misuse of pseudo-anonymity**

Because of the cryptographic character of blockchain, as well as the lack of a central authority, it is very easy for nodes to execute illegal or politically dangerous transactions via bitcoin in the darknet. Certain countries such as the US or Germany are attempting to implement regulatory steps to prevent the misuse of the technology and enforce its potential. For example, the Federal Government of Germany recognized the bitcoin currency as "private money", which can be used in multilateral accounts and transfers [30]. It should be noted that recent (May 2017) ransomware attacks demanded payment in bitcoins due to this very anonymity.

3.2 Hardware evolution

3.2.1 Data centre evolution

Data centres are at the centre of modern software technology, serving a critical role in the expanding capabilities of enterprises. The evolution of the data centre has passed through three stages: siloed data centres, virtualized data centres and software-defined data centres.

The siloed data centre relies heavily on hardware and physical servers, networks and storage. It is defined by the physical infrastructure, which is dedicated to a singular purpose and determines the amount of data that can be stored and handled by the data centre as a whole. This results in very low asset utilization at the price of high operational and capital expenditure. It can take an enterprise months to deploy new applications with a traditional data centre.

Between 2003 and 2010, virtualized data centres began to emerge, as the virtual technology revolution made it possible to pool the resources of the computing, network and storage operations of several formerly siloed data centres to create a central, more flexible resource that could be reallocated based on needs. By 2011, about 72% of organizations said their data centres were at least 25% [source: ZK research, 2013].

As technology is always pushing forward, resources are being pooled to create larger, more-flexible, centralized pools of computing, storage and networking resources. A major challenge in IT today is that organizations can easily spend 70% to 80% of their budgets on operations, including optimizing, maintaining and manipulating the environment. Likewise, how to handle dynamic workloads poses a significant challenge. Software-defined data centres combining server virtualization, software-defined networks (SDN), software-defined storage (SDS) and automation will enable the creation of a truly dynamic, virtualized data centre [31]. The control is exerted totally by software. That includes automating control of deployment, provisioning, configuring and operation with software, creating one centralized hub for monitoring and managing a network of data centres. Software also exerts automated control over the physical and hardware components of the data centre, including power resources and the cooling infrastructure, in addition to the networking infrastructure.

Worth noting in this context is the trend within data centres to shift to smaller, more agile (i.e. movable) data centres towards the edge. This includes, for example, "edge caching" approaches, adopted by companies such as Google to minimize latency in response, using scaled-down "out of the box" data centres, co-located or situated near to ISP nodes. This can be taken to its logical conclusion, if such caching/hosting is implemented at the base station level in wireless networks. Finally this trend can be extended to allow containerization at the base station and thus enable the docking of third party applications there.

3.2.2 IoT gateways/edge servers

As billions of end devices need to connect to the world, one of the most critical components of future IoT systems may be a device known as an IoT gateway. Traditional gateways have mostly performed protocol translation and device management functions. They were not intelligent, programmable devices that could perform in-depth and complex processing on IoT data. Today's smart IoT gateways are full-fledged computing platforms running modern operating systems (OSs), such as Linux and Windows. New generation IoT gateways are opening up huge opportunities to push processing closer to the edge, improving responsiveness and supporting new operating models.

The IoT gateway serves as an important bridge between operations and IT and also provides a cost-effective business model. By adding IoT gateways, the current field deployment could require no change in order to run new applications such as predictive maintenance on the gateway.

In scenarios such as smart manufacturing, with more and more robots, computer numerical control (CNC) machine tools and the like generating massive real-time data in the field, greater computing and storage resources will be necessary. In such situations a local cloud at the edge represents a good choice. More edge servers will be interconnected and provide pooled and scalable resources. Several software programmes or services can be deployed on an edge node simultaneously.

In order to fulfil these demands, the ECN should have enhanced computing, storage and networking capabilities, which requires more powerful hardware. Firstly, the central processing units (CPUs) on edge nodes should have greater power, higher frequency and larger L1 and L2 caches. Secondly, the edge nodes should have more extensive random access memory (RAM) and flash memories. Thirdly, in some cases, the edge nodes should be more capable of dealing with harsh environments, e.g. vibration, wide temperature variations, dust, electromagnetic interference in industrial sites and outdoor environments. Other requirements can include compactness and low power consumption. Enhanced hardware has increased the level of software that can be deployed at this level, from simple control-loop and communication software, to micro-kernel OSs to fully featured Windows and Linux kernels, with associated application software. This evolution has changed the function of the device from that of performing an information forwarding/protocol conversion role to being a sophisticated device in its own right with a hardware general diagram depicted in Figure 3-10 that is capable of:

- Security rule application on data streams, and a stateful packet inspection (SPI) firewall function as well as local encrypted operation
- Local business rule application and decision-making via ML
- Application docking through containerization

Much of this hardware evolution is driven by initiatives outside the industrial commercial sphere by amateur/hobbyist devices such Arduino and Raspberry PI. These affordable options have displaced many more expensive development boards for prototyping and are starting to find their way into final hardware designs.

3.2.3 Smart sensors/end nodes

In addition to the development of end devices, the higher end sensor devices have become substantial computing devices in their own right. It is an emerging trend that this device class is becoming sufficiently powerful not to require a gateway device, or indeed through application docking to function as a gateway themselves, and thereby reduce the role of a gateway purely to that of a firewall/security function. A couple of examples illustrate the concept.

- **Intelligent card reader**

Illustrating that a relatively simple device can also be a powerful computing tool, this device, depicted in Figure 3-11, is at its base a radio

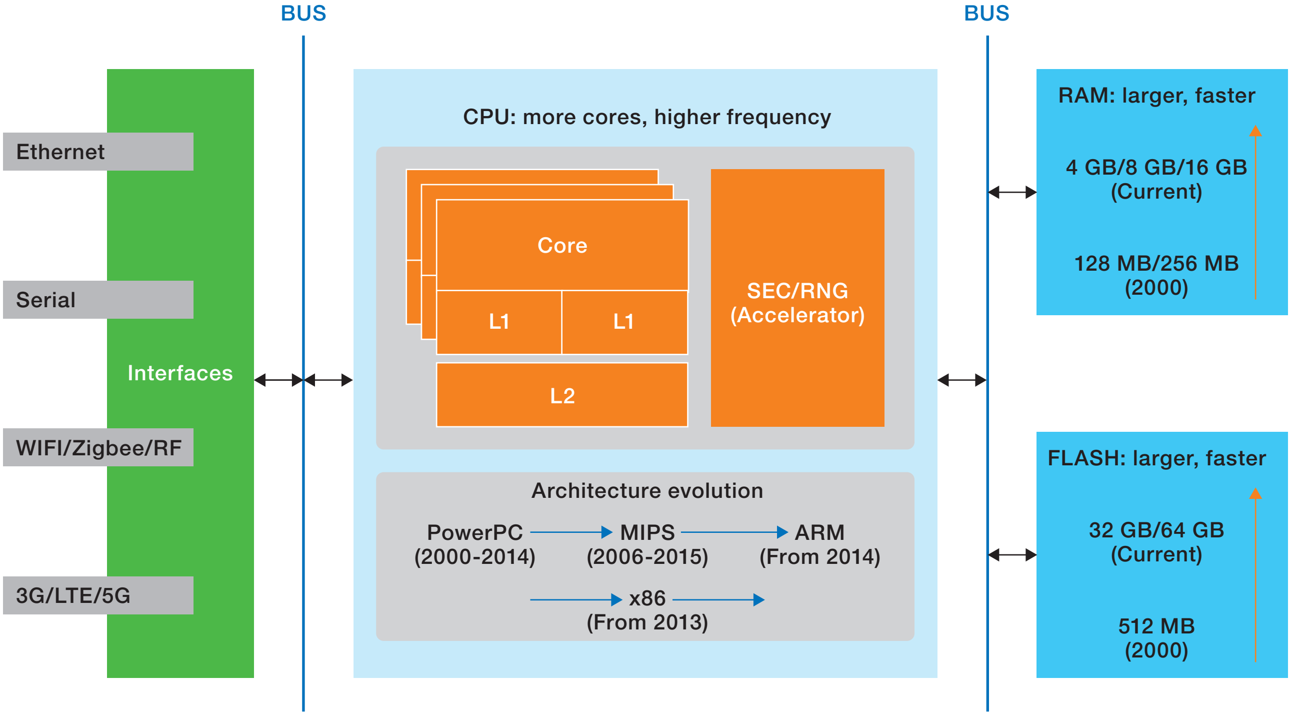


Figure 3-10 | Hardware diagram of ECN

frequency identification (RFID) proximity card reader but additionally has a number of inputs and outputs (4 inputs and 2 outputs), as well as RS485 and Wiegand inputs, and can establish its own connection via internet protocol (IP) with a remote server and channel not only its own local data (and cache it for independent operation-based on business rules) but also manage other devices of a lower order connected to it, including voice over IP (VoIP) communication. Therefore it performs the multiple functions of local sensor, edge server and gateway. The terminal also includes local application docking, which can be used to manage and gather data from other devices connected to its peripherals, as well as operate identity-based applications such as scheduling and biometrics.

- **High-end camera**

As well as performing its primary operation as a high-end IP camera, this camera, depicted in Figure 3-12, offers a powerful security and data-gathering capability, with dedicated video processors and core CPU capacity. It is equipped with analogue inputs and outputs and an application docking capacity, supporting both

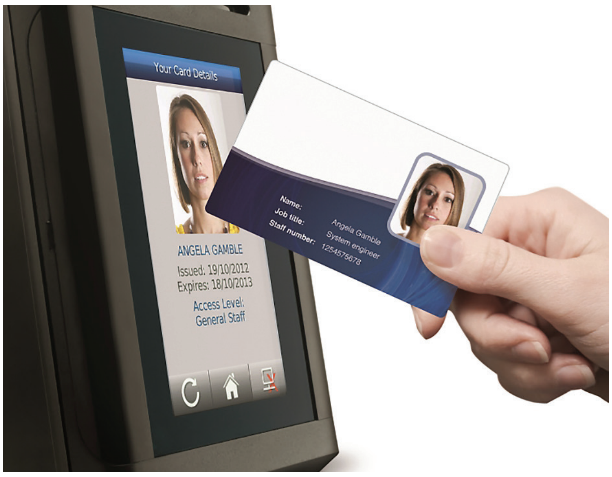


Figure 3-11 | Intelligent card reader

Figure 3-12 | Intelligent camera

input monitoring, output actuation and audio recording/streaming, again illustrating that within an edge node, a considerable level of computing power and the capability to control other, simpler, nodes is present.

3.3 Software evolution

3.3.1 IoT edge computing

Most of the IoT ECN technology, i.e. the technology which is the host for edge processing capabilities, runs on flavours of the Linux OS while using different kinds of processor architectures. An industry-wide trend is emerging to package edge computing capabilities into microservices and deploy them within containers on IoT ECNs, as illustrated in Figure 3-13. Containers provide security through isolation; they also serve as deployment units that simplify lifecycle management through less interdependency and complexity.

An exception to this Linux predominance is found in the manufacturing industry, where various versions of the Windows OS and Microsoft's .net platform make up the majority of implementations.

3.3.1.1 Device management

For energy-efficient IoT infrastructures, the end devices have to hold local policies concerning when to sense or when to connect to the network. For example, in the case of environmental parameters monitoring, the important sensing runs during the peak traffic hours when pollution might become a threat to public health. To convey device-specific capabilities, e.g. connectivity intervals, a new IoT tailored device management was standardized, called OMA Lightweight M2M (LWM2M).

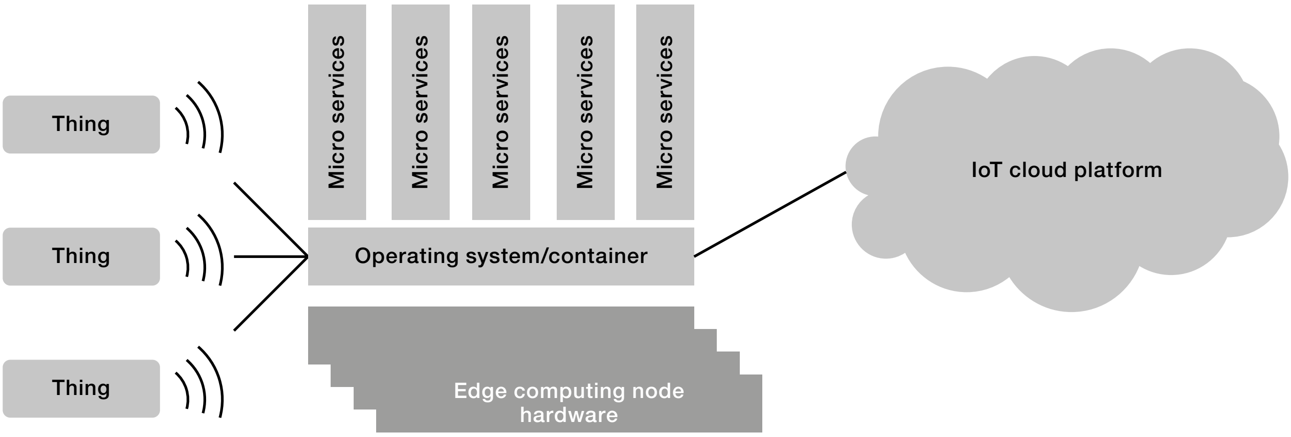


Figure 3-13 | Edge computing

Apart from location tracking, connectivity management and connectivity statistics, both firmware upgrade and software management of the software running on the device are supported by the protocol. Thus, the devices can be upgraded and can receive new capabilities from their management servers. For example, robots can enter into a new environment and, by receiving software packages containing the protocol adapters suited to communicate with the new environment, adapt their internal processing accordingly.

Being based on constrained application protocol (CoAP), the protocol is designed with the goal of interoperability and energy-efficient transport. Another transport that can be used for the data is short message service (SMS) or directly the subscriber identity module (SIM) IP connection. It also features store-and-forward features in which operations towards the end device are stored while the device is not reachable.

3.3.2 ECN OS

3.3.2.1 IoT gateway OS

In edge computing, data will be processed, analyzed and aggregated at the network edge near things or data sources. The IoT gateway is the perfect host of all these capabilities. The main edge computing functions of IoT gateway include cloud offloading, private data filtering, data aggregation, etc.

The OS running on IoT gateways is usually a general purpose OS such as Linux. Horizontal decoupling brings openness to gateways. Third-party applications can be deployed on gateway OS via a Host OS, a container (LXC, Docker) or a virtual machine.

The basic function of a gateway is always networking. The IoT gateway should support rich network protocols including L2 and L3 protocols such as virtual local area network (VLAN), routing protocols, multicast protocols and reliability protocols, so that the gateway can have flexible networking capabilities and can interconnect with equipments of third-party manufacturers. In order to meet the high performance and low latency requirements, the IoT gateway OS integrates various network protocols to general Linux, makes ameliorations to the data forwarding plane and selectively introduces hardware acceleration to some applications such as encryption/ decryption.

It is necessary to guarantee the security at OS level, including RAM and storage security, the secure operation environment of the host OS, containers and virtual machines, the security of encryption keys, anti-injection of malware, secure operation environment and malfunction isolation of third party applications, etc.

3.3.2.2 Lightweight OS for end devices

It is hard to develop intelligent hardware and the IoT applications due to the hardware resource constraint, the shortage of development platforms, the complex communication protocol, etc.

The innovative and business effective way is to provide an open and customizable platform.

The core parts of the platform, depicted in Figure 3-14, include:

- **Lightweight kernel**

  The OS kernel running on system-on-chips in tiny devices will hide the chipset difference between silicon companies, and this OS provides the drivers and reacts to events happening around the hardware. Considering the resource-constrained applications, the kernel has to have a light footprint, high start-up speed, low power consumption and fast response. The lightweight OS can be referred to a class of OSs including Tiny OS, Contiki, LiteOS, Mantis, etc.

- **Customizable services**

  Provides key services such as connectivity service, sensor manager, security engine, etc. which support agile development of new applications. The algorithm for each service is flexible and customizable. For example, the application scenarios might use the connectivities of Bluetooth Smart, 2G, 3G, LTE, Wi-Fi, etc. the stacks over these connectivity being customizable.

- **Open API**

  The point of the open API is to abstract the chaotic world of system on a chip (SoC) design and complexity of key services (such as connectivity) away from developers – leaving a cleaner, common interface to work with. Programmers who are handy with C++, JavaScript, HTML, Swift and other languages for phones, tablets and desktops can prototype and build applications for fiddly hardware ultimately hidden away under the platform. These programmers do not need to know about the undocumented registers and control bits and the rivalries plaguing the system-on-chip world. This is abstracted away by the open API, which tries to pave over the fragmentation.

Today's IoT largely exists in isolation, and it has been impossible to realize a truly interconnected world where devices are uninteroperable. The import of the lightweight OS platform to the vertical EI will open a new chapter. Since the platform provides the consistent APIs over the connectivity, security, application and other such domains, which hides the vendor's implementation differences, it is evident that it will fundamentally solve the interoperability issues of terminals (sensors), and terminals (sensors) with applications.

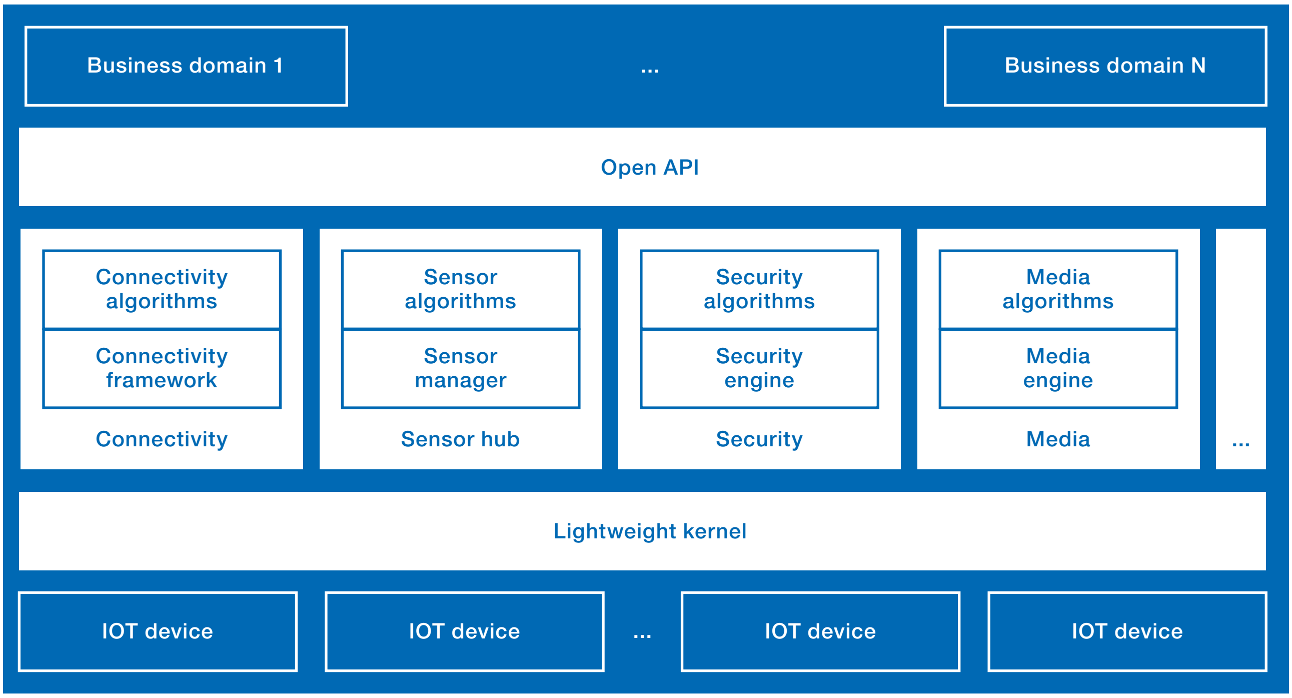


Figure 3-14 | Lightweight OS architecture [source: Huawei]

3.3.3 Containerization and microservices

A key architectural goal for edge computing is to have functionality isolation in the form of services which can be deployed anywhere, e.g. on a smart device, on an IoT gateway, in a micro data centre, inside a telecommunications network, in a company's data centre or in a public or private cloud.

This isolation can be achieved through the service concept, specifically microservices (which are totally self-contained) or pods (a Kubernetes concept designating a group of services sharing resources, e.g. a database).

The deployment of services requires a runtime environment that is the same across all possible deployment locations. There are three major options, as follows:

- Open Service Gateway initiative (OSGi) is a standardized runtime environment for Java (components written in other programming languages can be embedded). A (micro) service will be represented by an OSGi bundle.
- Virtual machines is a hardware abstraction. A (micro)service will be represented by the service functionality inside its own guest OSs.
- Containers is an OS abstraction. It provides all the dependencies, e.g. runtime libraries and components, of services.

The industry trend is favouring containers, as they are more lightweight than virtual machines (less resources consumption and faster start-up) and not tied to a specific programming language like OSGi.

3.3.3.1 Virtual machines

Virtual machines provide a software abstraction of computer hardware. This allows running multiple OS instances (also known as guest OSs) on a single piece of hardware. A virtual machine provides the mapping between the software interfaces and the actual hardware. It also manages the resources among these instances.

There is no restriction on guest OSs, for example various flavours of Linux and different versions of Windows can be run on the same hardware at the same time.

Virtual machines are being used on PCs as well as servers and provide benefits for a number of other scenarios.

The first scenario allows running for example Windows as a guest OS on a Linux or Mac OS natively hosted box, or vice versa.

Initially, virtual machines where used for consolidating servers on a smaller number of hardware systems. With the rise of cloud computing virtual machines played an important role for providing scalability and isolation.

Containers have become more popular in cloud computing as they have less overhead than virtual machines. They provide less flexibility, but cloud data centres typically provide standardized hardware running Linux, which removes the need for flexibility.

Virtual machines are also quite popular for quality assurance, as software can be tested on a variety of OS versions with different configurations and the core hardware re-used for different OS tests as required.

3.3.3.2 Containers

Containers provide virtualization on the OS level. Containers are user space instances executed in the user space of an OS kernel providing strong isolation between them. The OS kernel can provide resources management between different containers.

Unlike virtual machines, containers impose little or no overhead. However, containers are less flexible than virtualization, as only one OS is used, Linux being the most popular one.

Today containers are predominantly used within cloud platforms, providing horizontal scalability, isolation and resource and lifecycle management. Containers also play an important role in software engineering processes, as they allow developers to build software in a container which can be deployed in a production environment along with the container itself, simplifying, installing, providing maintenance and enhancing application security.

Docker has been the most popular container technology. However more recently rkt or Rocket has become widely used. The Open Container Initiative (OCI), a Linux Foundation project, aims to bring the competing container technologies together.

Kubernetes is another open source project, by the Cloud Native Computing Foundation (CNCF) for automating deployment, scaling and management of containerized applications. It was originally designed by Google for its data centre operations.

Kubernetes also introduces the concept of pods, which is a group of services deployed in containers that share certain resources, e.g. a database, and interact via interposes communication. Pods are an interesting concept for edge computing as they allow services to share scarce resources.

3.3.3.3 OSGi

The OSGi technology is a set of specifications that define a dynamic component system for Java. These specifications enable a development model where applications are (dynamically) composed of many different (reusable) components. The OSGi specifications enable components to hide their implementations from other components while communicating through services, which are objects that are specifically shared between components. This surprisingly simple model has far-reaching effects for almost any aspect of the software development process.

The OSGi has a layered model, which is depicted in Figure 3-15.

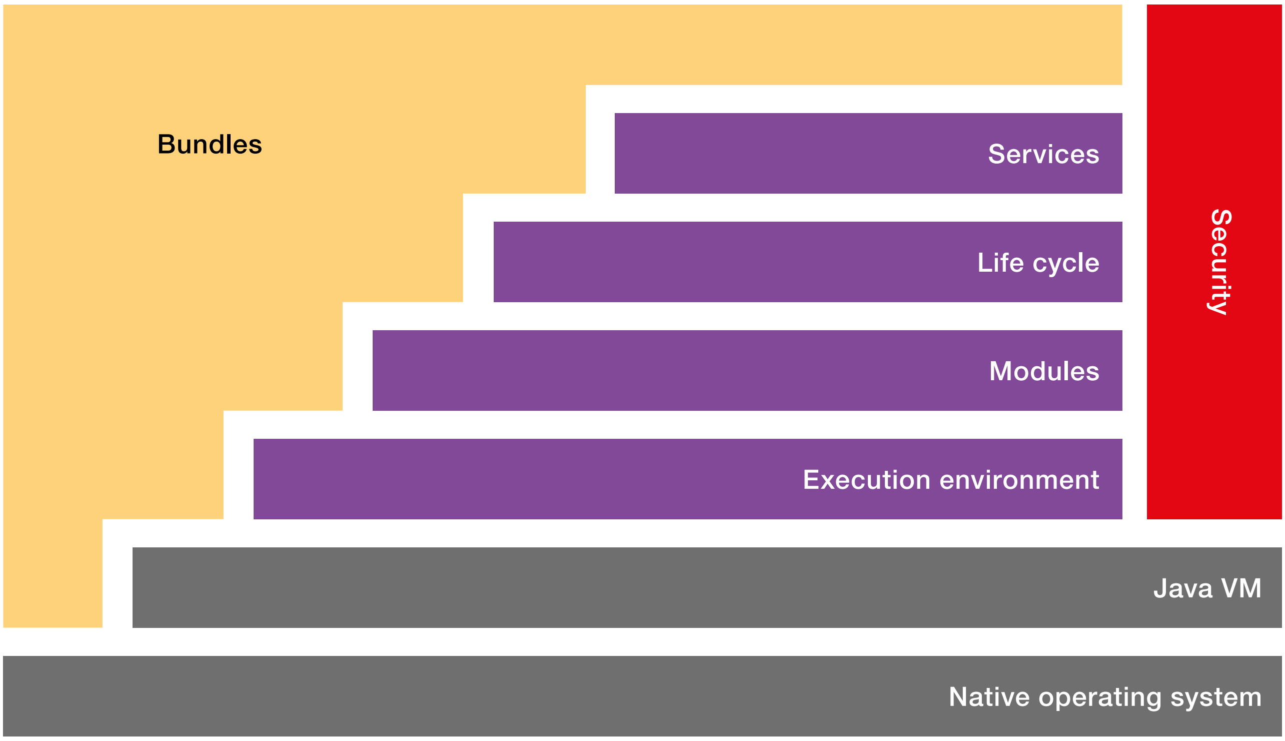


Figure 3-15 | Layered model of OSGi [source: OSGi]

The following list contains a short definition of the terms contained in Figure 3-15:

- Bundles: the OSGi components made by the developers
- Services: the layer that connects bundles in a dynamic way by offering a publish-find-bind model for plain old Java objects
- Life cycle: the API used to install, start, stop, update, and uninstall bundles
- Modules: the layer that defines how a bundle can import and export code
- Security: the layer that handles the security aspects
- Execution environment: defines what methods and classes are available in a specific platform

The fundamental concept underlying such a system is modularity. Modularity is about keeping things local and not sharing. It is hard to be wrong about things you have no knowledge of and make no assumptions about.

The services model is about bundles that collaborate. A bundle is a plain old Java archive (JAR) file: it can create an object and register it with the OSGi service registry under one or more interfaces. Other bundles can go to the registry and list all objects that are registered under a specific interface or class. This is depicted in the diagram in Figure 3-16.

OSGi provides support for EI/computing IoT solutions. It offers not only a component-oriented execution environment to dynamically maintain, manage, bill, and enhance networked devices and their applications remotely, but also various IoT-related standard components for developers to reduce time-to-market and development costs, such as device service, HTTP service, message queu telemetry transport protocol (MQTT) service, etc.

For example, an OSGi gateway can only install the bundles that are responsible for interacting with devices. Once a rule engine is needed on this gateway to trigger alarms in special situations, the rule engine bundles can be remotely deployed and installed on this gateway.

3.3.3.4 Containerization for the edge

As discussed above containers are the preferred technology for deploying pods and microservices in the cloud. Container technology is also very applicable to edge computing and early adopters already use containers or container-like technology on the edge.

It would be beneficial to the success of edge computing to provide a uniform runtime infrastructure that would allow to deploy services in the cloud as well as in the edge. This would allow the optimization of application and services between bandwidth and the location of storage

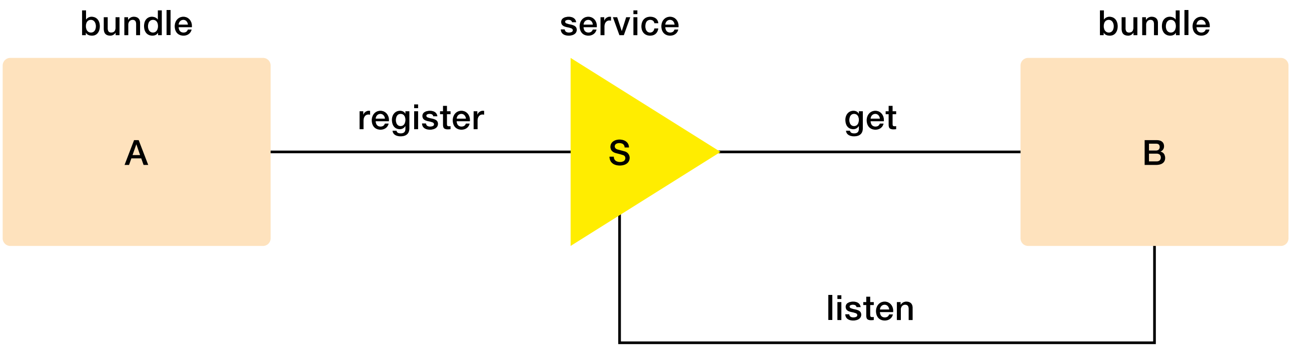


Figure 3-16 | Service of OSGi [source: OSGi]

and compute power for requirements such as latency, scale, reliability and, ultimately, cost.

A commonly accepted open source platform would reduce total cost of ownership for customers and would allow solution providers to focus on providing value-adding services and industry-specific solutions.

The Linux Foundation's Edge X Foundry project [32] is currently the most promising activity towards this unification.

3.3.4 Machine learning

3.3.4.1 Introduction

As outlined in *Basic Concepts in Machine Learning* by Jason Brownlee, traditional computer programming and data handling has been based on the encoding of rules, and generally the more cases covered the more successful the application could be considered to be. If the objective of programming can be considered to be automation, then ML is automating the process of automation. Instead of encoding rules, ML allows a framework to examine the data and discover the rules that underlie it, harnessed toward a particular objective, as can be seen in Figure 3-17.

Examples include predicting the sale price of a house based on a set of features (square feet, number of bedrooms, area, etc.), trying to determine if an image is that of a dog rather than a cat, or determining whether the sentiment of a set of restaurant reviews is positive or negative.

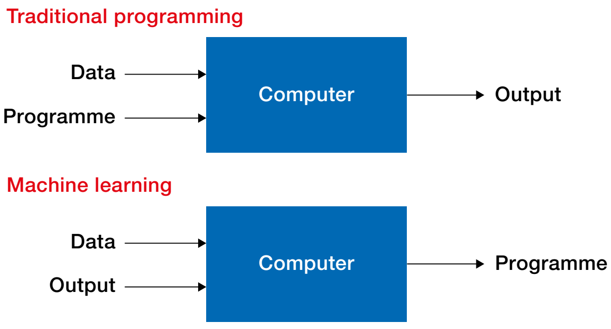


Figure 3-17 | Traditional programming vs ML

3.3.4.2 Some current applications of ML

Examples of current applications of ML include the following:

- Web search: ranking page based on what one is most likely to click on
- Computational biology: rational design drugs in the computer based on past experiments
- Finance: decisions on whom to send what credit card offers to; evaluation of risk on credit offers; how to decide where to invest money
- E-commerce: predicting customer churn; determining whether or not a transaction is fraudulent
- Space exploration: space probes and radio astronomy
- Robotics: how to handle uncertainty in new environments; autonomous machines; self-driving cars
- Information extraction: asking questions from databases across the web
- Social networks: data on relationships and preferences; ML to extract value from data

3.3.4.3 Key elements of ML

There exist tens of thousands of ML algorithms, and hundreds of new algorithms are developed every year.

Every ML algorithm has three basic components:

- Representation: how to represent knowledge; examples include decision trees, sets of rules, instances, graphical models, neural networks, support vector machines, model ensembles and others

- Evaluation: the way to evaluate candidate programmes (hypotheses); examples include accuracy, prediction and recall, squared error, likelihood, posterior probability, cost, margin, entropy, Kullback-Leibler divergence and others
- Optimization: the way candidate programmes are generated, known as the search process; for example, combinatorial optimization, convex optimization, constrained optimization

All ML algorithms are combinations of these three components.

3.3.4.4 Types of learning

There are four types of ML:

- Supervised learning (also called inductive learning): training data includes desired outputs
- Unsupervised learning: training data does not include desired outputs; an example is clustering
- Semi-supervised learning: training data includes a few desired outputs
- Reinforcement learning: rewards result from a sequence of actions; this appeals to AI practitioners, it is the most ambitious type of learning

Supervised learning is the most mature, the most studied and the type of learning most used by ML algorithms. Learning with supervision is much easier than learning without supervision. Some key concepts involved:

- Classification: when the function being learned is discrete
- Regression: when the function being learned is continuous
- Probability estimation: when the output of the function is a probability

The next step beyond ML involves a complementary area called artificial intelligence (AI), which leans more on methods such as neural networks and natural language processing that seek to mimic the operation of the human brain.

3.3.4.5 Potential applications in IoT

The advantages of a particularly supervised learning are clear in IoT applications – for sensor applications (including audio and video), the training data set can be generated and models refined offline (in the cloud for example), with the refined model then being loaded at the sensor level. There are clear advantages to this approach:

- Core software updates are reduced; rules and operating code are separated and updates limited to the model data only.
- Security is enhanced, as the potential for update-based attacks is reduced and the security exposure of potential security weaknesses in incremental updates limited.
- Rollback and rollforward of updates is simplified.
- Model comparison can be performed at the CPU level, if the hardware supports it.

3.3.4.6 Current state of the art

Like many other IoT-enabling technologies, however, machine intelligence (MI) research and development has largely been restricted to the IT sector, as the complexity of convolutional neural networks (CNNs), hidden Markov models (HMMs), natural language processing and other disciplines used in the creation of ML algorithms and deep neural networks (DNNs) requires storage and computing resources usually only accessible on a data centre scale.

One of MI's early excursions into the OT space came with the release of the NVIDIA Jetson TK1 platform in 2014. Based on the Tegra K1 SoC and its 192-core Kepler graphics processing unit (GPU) and quad-core ARM Cortex-A15, the

Jetson TK1 brought data centre-level compute performance to computer vision, robotics and automotive applications, but also provided embedded engineers with a development platform for the CUDA deep neural network (cuDNN) library. The cuDNN primitives enabled operations such as activation functions, forward and backward convolution, normalization, and tensor transformations required for DNN training and inferencing, and the combination of this technology with the Jetson TK1's 10 W power envelope meant that deep learning frameworks such as Caffe and Torch could be accessed and executed on smaller OT devices. In addition to high cost embedded approaches, pattern matching low level instructions in embedded processors have been added (e.g. Intel Quark).

3.3.4.7 Impact on user experience

The best user interface is no user interface. As IoT makes manual data input largely obsolete, and ML and AI take over decision-making, this maxim is becoming a reality.

How is the UI eliminated? A computing system's UI mainly serves two purposes (excluding entertainment):

- Enabling data input
- Providing information for human decision-making

Both functions are becoming largely obsolete for the following reasons:

- IoT enables direct data input from the source into the computer system.
- AI automates the decision-making.

The human user may only need to be involved in setting policies and providing feedback, and in exceptional situations. But even the latter may be only a transitional phase, as the discussion about eliminating pedals and the steering wheel, the UIs, from self-driving cars indicates. Thus, the best UI is no UI.

3.4 Architecture

3.4.1 Data collection architectures

3.4.1.1 Traditional data gathering model, multidrop/analogue sensors

The multidrop network depicted in Figure 3-18 is the prevailing network model among the majority of systems currently deployed, shared by fire, security and analogue camera systems. The lowest level of compute is at the gateway level, and often this is purely a forwarding function to a central server. It is true to say that very few instances of new equipment will follow a model of this type. The exception to the assertion above may be in process control and factory automation functions.

3.4.1.2 Current data gathering model, wireless/networked sensors

The wireless sensor network, see Figure 3-19, is the prevailing network model among the majority of systems currently sold, shared by fire, security

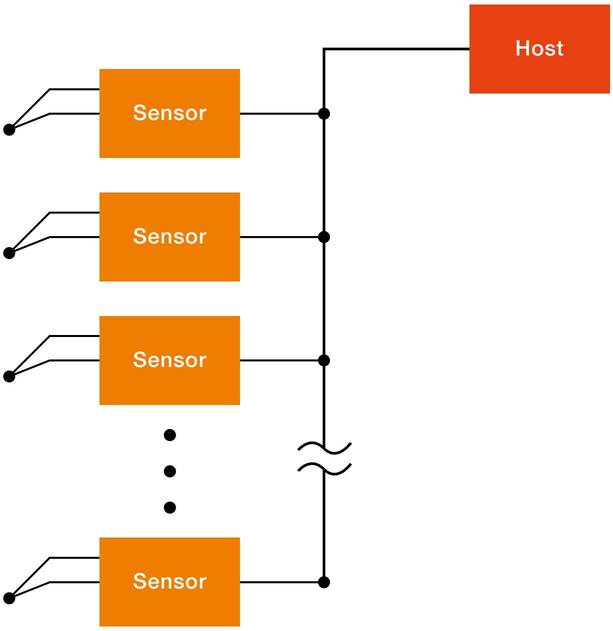


Figure 3-18 | Multidrop network

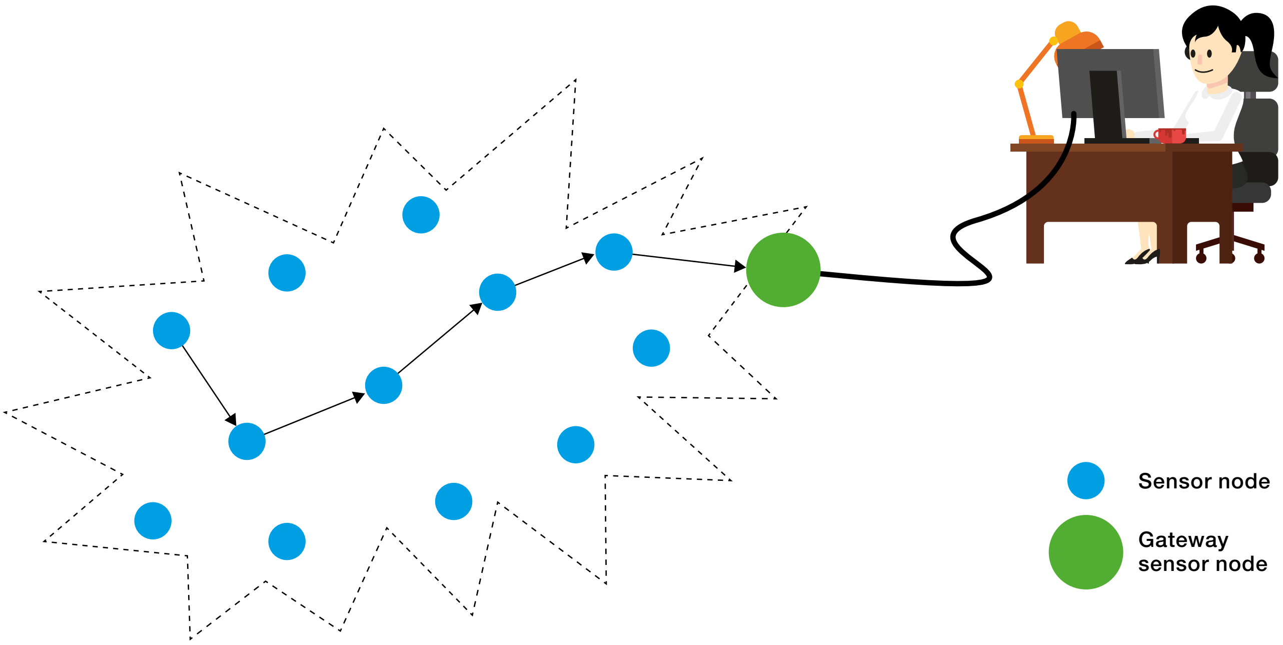


Figure 3-19 | Wireless sensor network

and digital camera systems. The lowest level of compute is at the gateway level, and often this is purely a forwarding function to a central server, and onwards to the cloud.

3.4.1.3 Current/future data gathering model, intelligent sensors

There are two additions to this current scenario that will change both the data and the analysis rapidly. The first is intelligent sensors that are able to understand what the expected range of data they are collecting should be. If the data is in the expected range, the sensor does not initiate a report, whereas it reports on unexpected readings. The other innovation that is taking hold is that of mesh networks for the sensors. Instead of the sensor reporting to a single device through a single connection, the sensor now reports to a mesh. The mesh is able to route data quickly and effectively autonomously. Mesh sensors combined with intelligent sensors make for interesting combinations, including a resilient network that houses and manages sensors that only report when there are variances outside the norm. The result of this is called the intelligent sensor network, functioning according to the diagram depicted in Figure 3-20.

3.4.2 oneM2M

Formed in 2012 by eight of the world's leading ICT standards development organizations, notably ARIB (Japan), ATIS (US), CCSA (China), ETSI (Europe), TIA (US), TSDSI (India), TTA (Korea) and TTC (Japan), oneM2M is a global Standards initiative which defines a single horizontal service platform for exchanging and sharing data among IoT applications that can be used in various vertical industries, including smart home, healthcare, transportation and manufacturing.

Figure 3-21 illustrates the oneM2M functional architecture [33], which comprises three layers: an application layer, a common services layer and

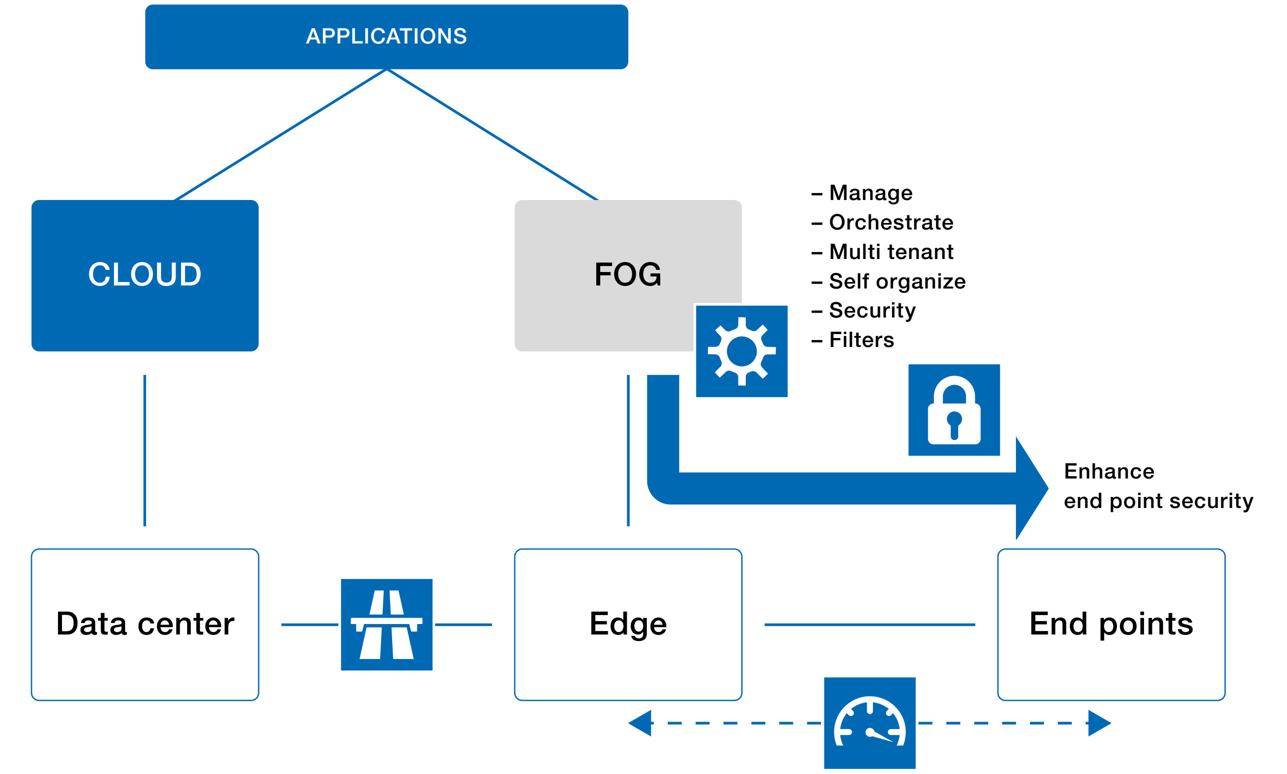


Figure 3-20 | Intelligent sensor network

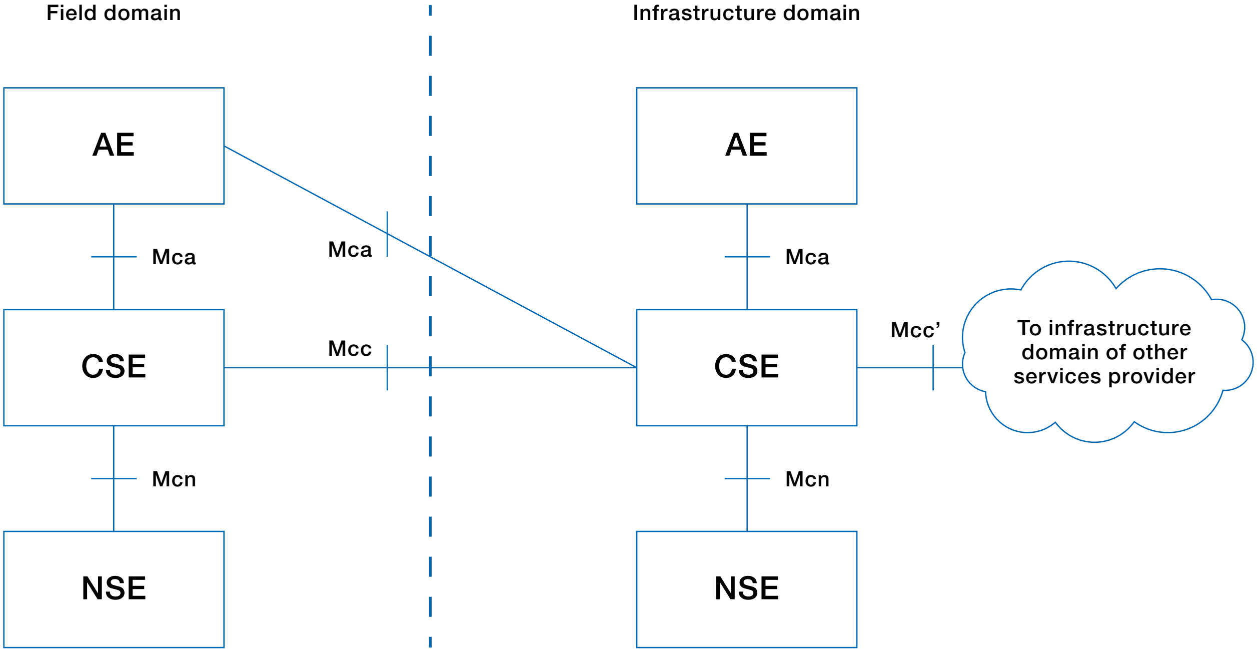


Figure 3-21 | oneM2M functional architecture [33]

the underlying network services layer. For each layer, the logical functions are represented by the corresponding entities:

- Application entity (AE): is an entity in the application layer that implements an M2M application service logic. Examples of AEs include an instance of a fleet-tracking application, a remote blood sugar monitoring application, a power metering application, or a controlling application.
- Common services entity (CSE): represents an instantiation of a set of "common service functions" of the M2M environments. Such service functions are exposed to other entities through the Mca and Mcc reference points. Reference point Mcn is used for accessing underlying network service entities. Examples of service functions offered by CSE include: data management, device management, M2M service subscription management, and location services. Such "sub-functions" offered by a CSE may be logically and informatively conceptualized as common services functions (CSFs).
- Underlying network services entity (NSE): provides services from the underlying network to the CSEs. Examples of such services include device management, location services and device triggering. No particular organization of the NSEs is assumed.

Each of the oneM2M functional entities may reside on both the field domain and the infrastructure domain depending on the deployment configurations, e.g. whether the devices in the field domain are resource constrained or whether one or more intermediate gateways are needed for distributed data storage, local process or interworking. Such configuration options are illustrated in Figure 3-22, where the concept of "node" is introduced as the logical equipment that may contain AE and/or CSE. The illustration does not constrain the multiplicity of the entities nor require that all relationships shown are present.

Depending on the deployment location and whether it contains oneM2M AEs and/or CSEs, several types of oneM2M nodes are defined:

- Infrastructure node (IN): is a node in the infrastructure domain that contains one CSE and zero or more AE. A physical mapping of an IN could be an IoT platform. Therefore, a CSE in an IN may contain CSE functions (as the server role) not applicable to other node types (as the client role).
- Application service node (ASN): is a node in the field domain that contains one CSE and at least one AE. A physical mapping of an ASN could be an IoT device.
- Application dedicated node (ADN): is a node in the field domain that contains at least one AE and does not contain a CSE. A physical mapping of an ADN could be a constrained IoT device.
- Non-oneM2M device node (NoDN): is a device node in the field domain that does not contain oneM2M entities (neither AEs nor CSEs). Such nodes represent devices attached to the oneM2M system for interworking purposes, including management.
- Middle node (MN): is a node in the field domain that contains one CSE and contains zero or more AEs. A MN is different from an ASN in that a MN (MN-CSE) can communicate with both IN and ASN/ADN/NoDN on two sides. It is the intermediate entity that can provide proxying functionalities and local process (EI). A physical mapping of an ADN could be an IoT Gateway. oneM2M architecture supports concatenate MNs. Note that CSEs resident in different nodes can be different and are dependent on the services supported by the CSE and the characteristics (e.g. different memory, firmware) of the physical entity that contains the CSE's node. oneM2M CSEs provide services referred to as CSFs. CSFs provide services to the AEs via the Mca

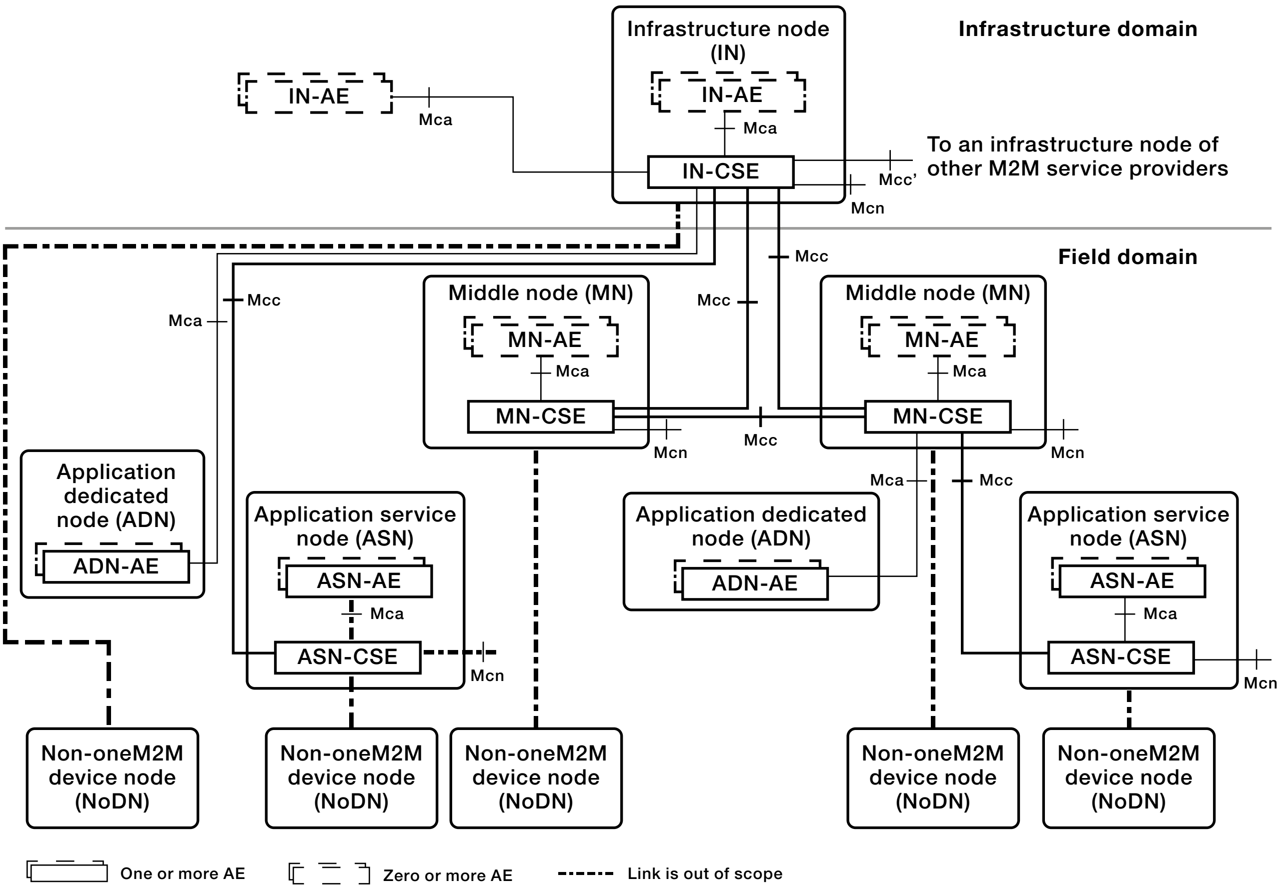

Infrastructure node (IN)
Infrastructure domain
IN-AE
Mca
IN-AE
Mca
IN-CSE
To an infrastructure node of other M2M service providers
Mcc'
Mcn
Mcc
Field domain
Mca
Mcc
Middle node (MN)
MN-AE
Mca
MN-CSE
Mcc
Middle node (MN)
MN-AE
Mca
MN-CSE
Mcn
Mcc
Mcn
Application dedicated node (ADN)
ADN-AE
Application service node (ASN)
ASN-AE
Mca
ASN-CSE
Mcn
Mca
Mcc
Application dedicated node (ADN)
ADN-AE
Application service node (ASN)
ASN-AE
Mca
ASN-CSE
Mcn
Non-oneM2M device node (NoDN)
Non-oneM2M device node (NoDN)
Non-oneM2M device node (NoDN)
Non-oneM2M device node (NoDN)
Non-oneM2M device node (NoDN)
One or more AE
Zero or more AE
Link is out of scope

Figure 3-22 | Configurations supported by oneM2M architecture [33]

reference point and to other CSEs via the Mcc reference point. An instantiation of a CSE in a node comprises a subset of the CSFs as illustrated in Figure 3-23.

oneM2M adopts the RESTful architecture so that all services provided via Mca or Mcc are based on the typical create, retrieve, update, delete, notify (CRUDN) operations which can be bound to different protocols (RESTful or non-RESTful) such as HTTP, CoAP, MQTT and WebSocket. The choice of the protocol bindings is dependent on implementation.

Note that oneM2M does not specify the Mcn reference point and is independent from specific network technologies. However, the CSEs may interact with the NSE via the Mcn based on the network APIs specified by other organizations (e.g. OMA or 3GPP) so as to leverage and coordinate with the underlying network services to provide better services to the IoT applications.

oneM2M architecture provides native support for EI/computing thanks to the flexible deployment configurations as shown in Figure 3-23. Given the higher capacity of the storage and computing power, MNs (IoT gateways) and ASNs (smart IoT devices) are typically the ideal hosts for the EI.

A MN/ASN can host both CSEs and AEs (see Figure 3-23). CSEs can provide local/edge services like data storage, resource discovery, device and application management, communication buffering, while one or more AEs (local applications) can do additional versatile

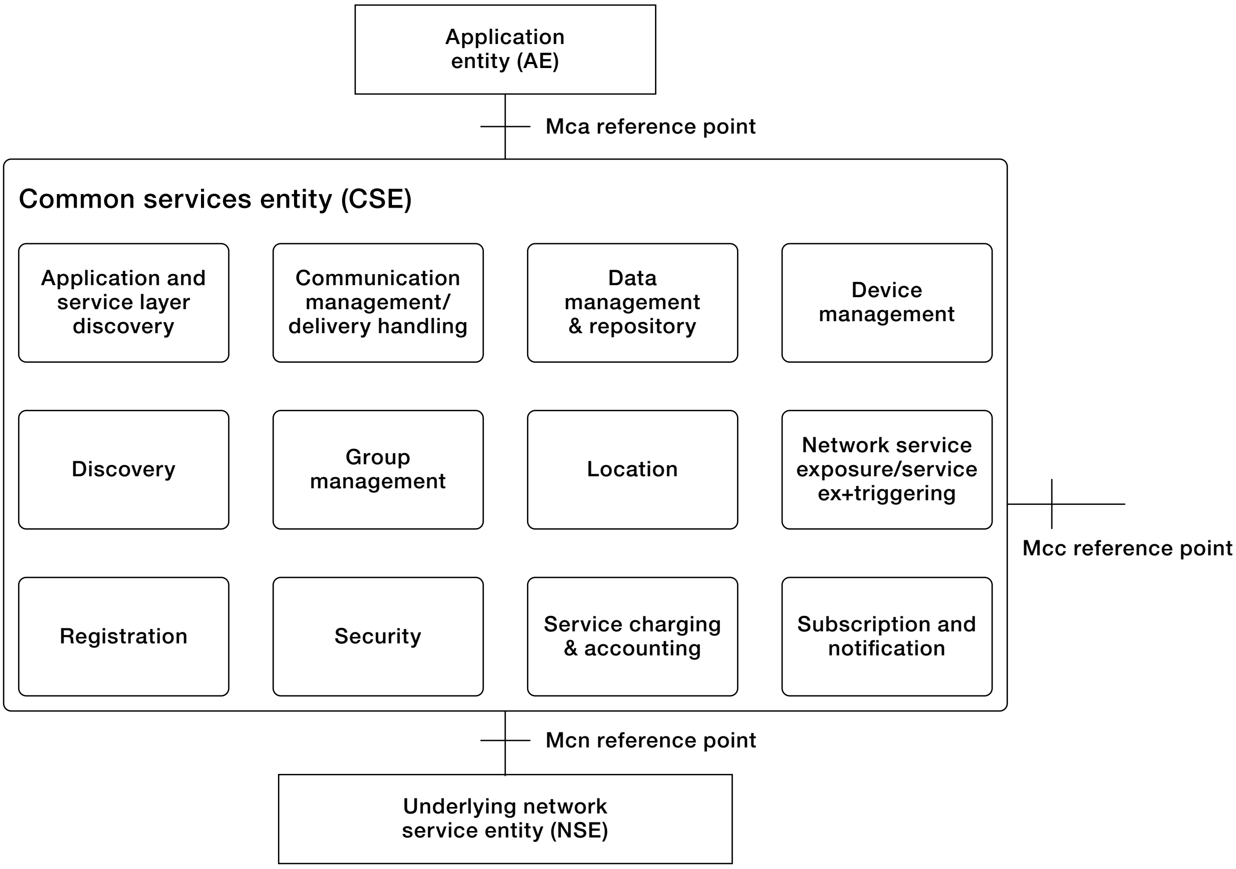

Application entity (AE)
Mca reference point
Common services entity (CSE)
Application and service layer discovery
Communication management/ delivery handling
Data management & repository
Device management
Discovery
Group management
Location
Network service exposure/service ex+triggering
Mcc reference point
Registration
Security
Service charging & accounting
Subscription and notification
Mcn reference point
Underlying network service entity (NSE)

Figure 3-23 | oneM2M common services functions [33]

process (edge computing) on the service data, including local data filtering/mining, distributed rule execution, mash-ups, etc.

Some AEs on MN/ASN may also perform protocol translation and data model mapping as the interworking proxy entities (IPE) so that heterogeneous non-oneM2M IoT devices and area networks (like Zigbee, OCF) can be in the oneM2M system to enable flexible system integration.

Moreover, MNs and ASNs can communicate with each other as well as with ADNs/NoDNs in a local area network for fast data process and reaction without always going through the cloud (i.e. the IN) in order to save bandwidth, storage and latency.

For example, an application (MN-AE) on a home gateway (MN) may collect the room temperature data from a temperature sensor (ADN), then trigger the air conditioner (ASN) to start heating based on a configurable automation rule. Such simple tasks do not have to go through the network, which is not efficient and may cost extra expense to the user.

Another case could be a smart factory gateway (MN) which collects time-series diagnostic data from a pipeline (ASN/ADN), stores it in local storage (MN-CSE) and backlogs it (with compression and filtering) to the cloud (IN) daily or weekly as a routine job. In the meanwhile, the local application (MN-AE) on the gateway (MN) can also do predictive analysis based upon the local data in real-time. If an abnormal pattern is detected or predicted, it will issue an emergent report to the cloud (IN) with the full raw data, if necessary. Communication bandwidth from the edge to the cloud as well as the cloud storage can be greatly saved from low-value data reporting unless an emergency situation occurs. Furthermore, the local application (MN-AE) can also take immediate countermeasures, e.g. sending a control command directly to the pipeline to shut down or slowdown, to avoid catastrophic damage or loss that can occur in milliseconds before the emergency can be reported to the cloud or the instructions can be received from the cloud.

Similar cases can also be found in self-driving. Connected cars (MN or ASN) will perform most of the computing tasks locally, e.g. adaptive cruise control, collision avoidance, while downloading traffic information from the cloud (IN) from time to time, or report its location and damage to the cloud (IN) only in case of an accident.

3.4.3 IIC architecture

The industrial internet is an internet of things, machines, computers and people, enabling intelligent industrial operations using advanced data analytics for transformational business outcomes. The Industrial Internet Consortium (IIC) reference architecture, following the OMG/ODP tradition and the IEC guidelines, considers the four viewpoints: 1v) Business, 2v) Usage, 3v) Functional, 4v) Implementation. The Functional viewpoint in turn considers five domains: 1d) Control, 2d) Operations, 3d) Information, 4d) Application, 5d) Business, supported by the six common security functions: audit, identity, cryptography, privacy, authentication, and physical protection. This three-tier model is one representative of the implementation view. Figure 3-24 describes the three-tier model [34], in which 1d is mainly at the edge tier, 2d and 3d

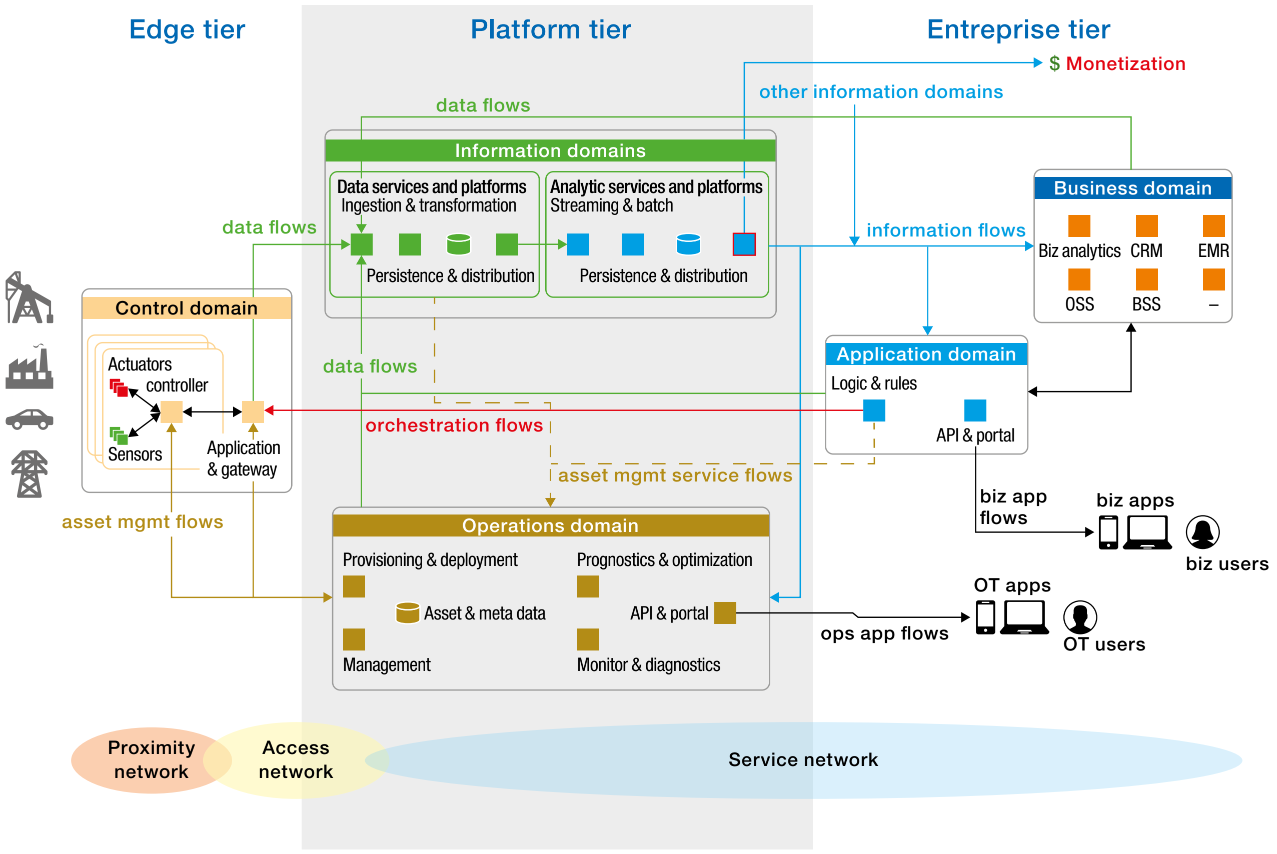


Figure 3-24 | IIC architecture [source: IIC]

are mainly at the platform tier, and 4d and 5d at the enterprise tier. Consequently EI requires migration of some of 3d and 4d to the edge domain. Intelligence, along with security, resilience, analytics, etc., is considered by IIC as a key system concern; it relates to the highest, i.e. the third, level of understanding in communication, meaning that it facilitates the interpretation of the sender's intent.

Intelligence is to be supported in several places, for example intelligent decisions can be supported by the automated service discovery. However the key placeholder of the IIC intelligence is intelligent and resilient control (IRC). The IIC architecture document sketches a number of modelling considerations relevant to the IRC design, along with a sample IRC workflow, and maps them to the functional components such as planners, predictors, blame assigner and ethical governor.

3.4.4 OPC-UA architecture

There are clear indications that the digital transformation will take place across all industrial domains, and at the same time a number of technical challenges will arise. The key areas of interest connectivity, communication and data exchange directly build upon one another to ensure cross-domain interoperability. Based on an extended conceptual open systems interconnection (OSI) model, this starts from the physical layer and ends on semantic-based exchange of knowledge.

Developed within the international OPC Foundation, a global non-profit organization with around 500 members from system integrators and industrial suppliers, the Open Platform Communications Unified Architecture (OPC-UA) middleware, see Figure 3-25, is mainly being used in the automation industry. Since 2006 it is the successor to the former Object Linking and Embedding for Process Control (OPC) architecture and defines two different communication types: either directly exchanging binary data using raw transmission control protocol (TCP) sockets or exchanging extensible markup language (XML) data via simple object access protocol (SOAP) and HTTP over TCP, with the architecture defined in [35]. Since June 2016 further application-level publish/subscribe protocols such as advanced message queuing protocol (AMQP) are being evaluated. The Standards further define common base services such as (historical) data access, alarms and conditions and programmability, as well as a common object-oriented metamodel for describing exchanged information.

3.4.5 Core networks

3.4.5.1 4G core network

For the 4G access network, the 3GPP standardized the evolved packet core (EPC), see Figure 3-26, as the all-IP core architecture providing connectivity functionality for the 3GPP family of access technologies, including LTE. EPC includes convergent mechanisms for:

- authentication and authorization access to 3GPP and non-3GPP access networks having the home subscriber server and the

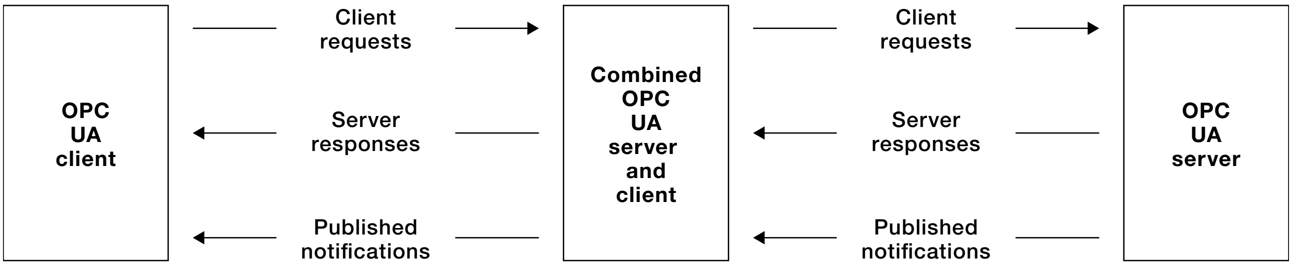


Figure 3-25 | OPC-UA system architecture [source: OPC-UA]

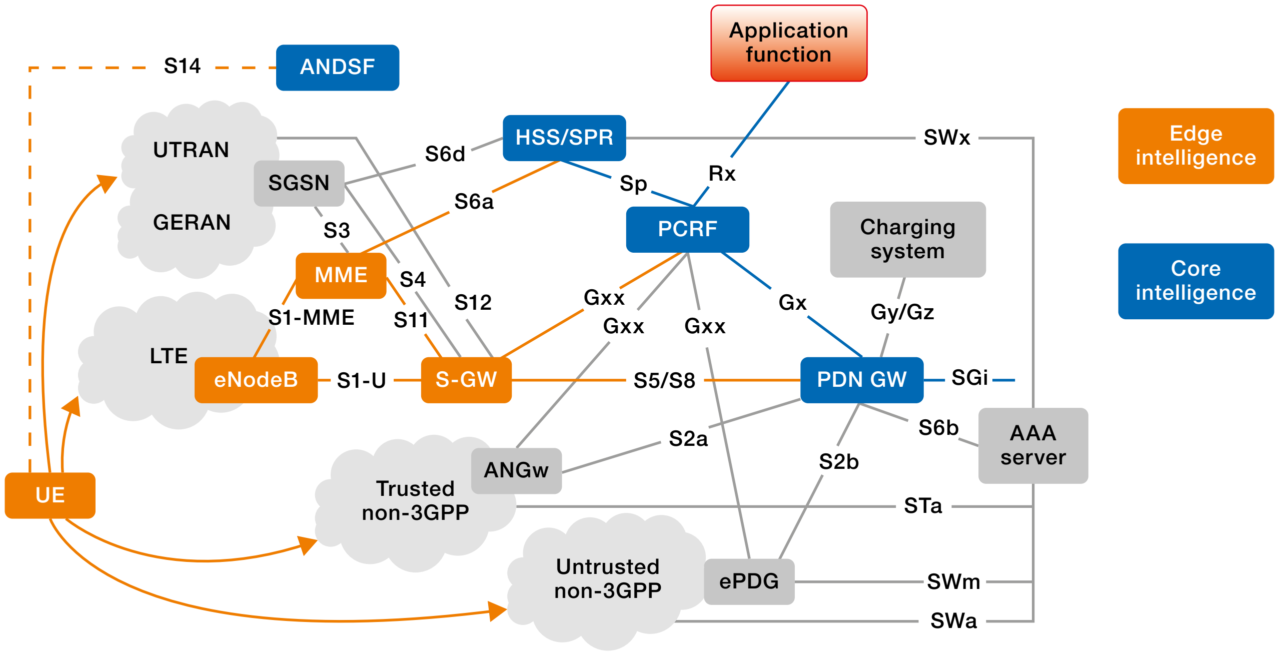


Figure 3-26 | EPC simplified architecture

authentication, authorization and accounting (AAA) server [36], holding the identity of the user endpoints (UE) and authenticating them for the two types of access networks;

- mobility support, transparent to the correspondent nodes. The mobility management entity (MME) preserves the connectivity sessions and the packet data network gateway (PDN-GW) anchors the UE data traffic and ensures the transparent mobility as well as the convergent accounting;
- the policy-based resource reservations quality of service (QoS) and charging, controlled by the policy and charging rules function (PCRF). The application function (AF) is communicating with the PCRF in order to request QoS capabilities for a specific application session;
- the access network discovery and selection function (ANDSF) is capable of sending recommendations/policies to the end devices for selecting a specific access network at a certain location and time interval. Thus, the UE can adapt its network usage based on location, coverage, price, available bandwidth of the advertised networks and the current required QoS. This functionality can be used by the network operator to offload the traffic at certain locations and time slots according to predictions or simple statistics.

The control plane of the EPC core network is constituted by the HSS, PCRF, ANDSF, MME and PDN-GW for the respective capabilities: authentication, QoS and charging, device connectivity management and mobility. These components are the entry points for intelligence in the network, if we consider them as being flexible to changing policies and adaptive to the monitored network behaviour.

Considering the mobility support, starting from monitoring information about service usage, patterns of both human and machine UE behaviour can be learned in terms of what kind of services need handovers between base stations (evolved Node B) eNodeBs)) or between access networks

(LTE and other non-3GPP networks). This acquired intelligence can then be used by the system to pre-allocate resources on the edge, for example scaling up the MME or serving gateway (SGW) components before a peak time occurs at a certain location (either there are a lot of users attaching or they are localized and using media services intensively).

Going a step forward, for high mobility applications that need low delay communication, 3GPP has defined the mechanism called local breakout, in which the application server is deployed in the back of the eNodeB directly. The communication is shortcut and is no longer forwarded to the SGW and PDN-GW and then to the internet via the SGi interface. This allows for even more intelligence to be placed on the ECNs of the 4G network to accommodate QoS and minimum packet loss from the established sessions for applications with high mobility and low delay communication as requirements (see section 3.4.7).

3.4.5.2 Software networks and software-defined networks

By adopting the paradigm to deploy network functions (NF) as programs on top of common off-the-shelf hardware, the main focus of the technology completely shifted towards the dynamicity offered by software programmes, resulting in software networks. Being the most customizable form of control, software programmes represent the full convergence between the telecom and IT industries unleashing new forms of innovation.

In the world of software-defined networking, the intelligence resides in the SDN controller that informs one or multiple SDN switches how to route packets, see Figure 3-27. The SDN switches are programmable switches that interface via a protocol with the SDN controller. Monitoring of the data flows is also supported so that the SDN controller can be informed about the traffic the SDN switches are handling. One of the SDN protocols supporting these features is OpenFlow [37].

The deployments of SDN networks started with campus networks and very fast device manufacturers and operators modified the EPC architecture by separating the control plane (CP) from the user plane (UP) having the SDN controller connected to the MME-SGW-PGW components and sending rules towards the layer of SDN switches [38].

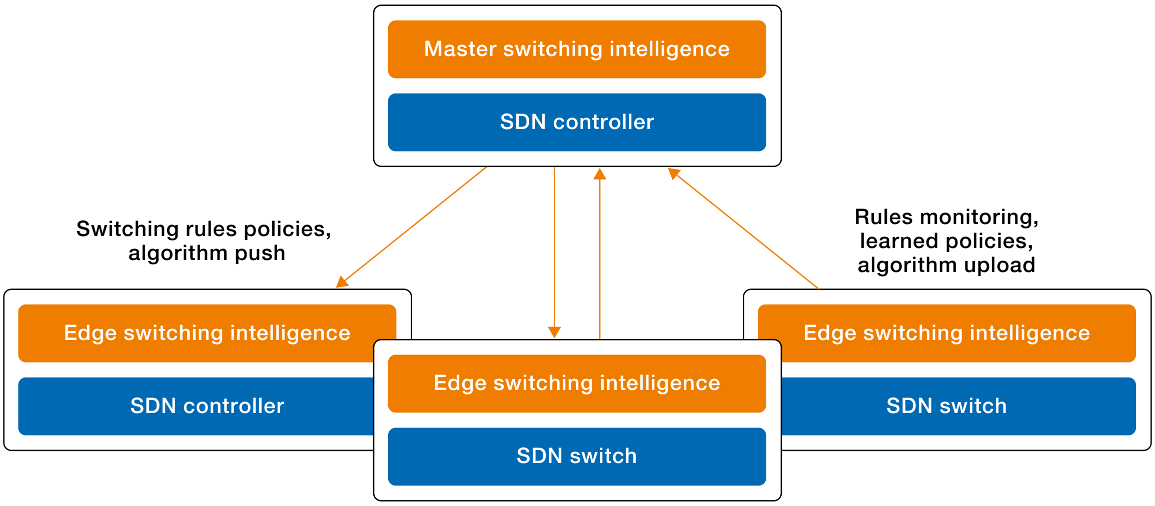

Master switching intelligence
SDN controller
Switching rules policies, algorithm push
Rules monitoring, learned policies, algorithm upload
Edge switching intelligence
SDN controller
Edge switching intelligence
SDN switch
Edge switching intelligence
SDN switch

Figure 3-27 | SDN and EI

The flexibility of the software-defined networks is that initial switching algorithms or policies can be provided to the switching layer and, based on rules/policies/algorithms provided by the edge nodes, the master switching entity can adapt and improve the edge switching intelligence. The SDN switches can receive algorithms to process the data, and based on the traffic that is monitored can improve the algorithm and upload it to the SDN controller.

It is believed that applying SDN to edge computing enables millions of devices to access the network and supports flexible scalability. SDN provides highly efficient and low-cost automatic operations and maintenance (O&M) and realizes policy collaboration and convergence of networks and security.

3.4.5.3 Network function virtualization and orchestration

The Standard ETSI network function virtualization (NFV) introduces a new approach to virtualizing networks and services in order to "simplify the roll-out of new network services, reduce deployment and operational costs and encourage innovation". Thus, multiple network operators (tenants) can deploy customized network services with different virtual network functions (VNFs) on a common infrastructure, thus realizing network sharing. By composing a set of NFs into a topology, a network service can be obtained that is fulfilling the service functional and behavioural specification. VNFs can be chained with other VNFs and/or physical network functions (PNFs) to realize a network service.

Some of the current use cases of ETSI NFV extracted from [39] include:

- Mobile core network
- IP multimedia subsystem
- Virtualization of the base station, home environment
- Content delivery network (CDN)
- Virtual private networks (VPN) as a service
- Fixed access network virtualization

An important component in the ETSI NFV specification is the NFV management and orchestration (MANO), see Figure 3-28 [39]. This component is capable of using VNF managers to deploy, monitor and determine the lifecycle of VNFs on virtualized infrastructures such as OpenStack [40] or Amazon Cloud. The operation and business support systems (OSS and BSS) and element management system (EMS) perform classical network management tasks, such as fault, configuration, accounting, performance and security (FCAPS) management.

From the EI point of view, NFV can be used to automatically scale up or scale down (deploy new instances or shut down instances of) services so that the system as a whole can cope with local requirements such as peak usage or energy efficiency.

When designing an NFV deployment at the edge, knowledge of the state of the virtual environment is necessary in order to decide whether the resources of Compute, Memory and Network are sufficient to support a needed adaptation of the services. That information might not suffice in practice, as studies have identified scenarios in which the real usage of the virtual resources might be influenced by the services running, e.g. "noisy neighbor" [41]. A solution for improving this situation is to first monitor the real state of the virtual resources and the impact on the system behaviour. Monitoring information obtained for example from injected automated tests reporting the current delay of service establishment, e.g. session establishment or media exchange, is one of the strategies, especially when there are no available statistics on running a specific service on a specific virtualized and shared infrastructure. If the measured delay is still acceptable, the system running the service can become intelligent and

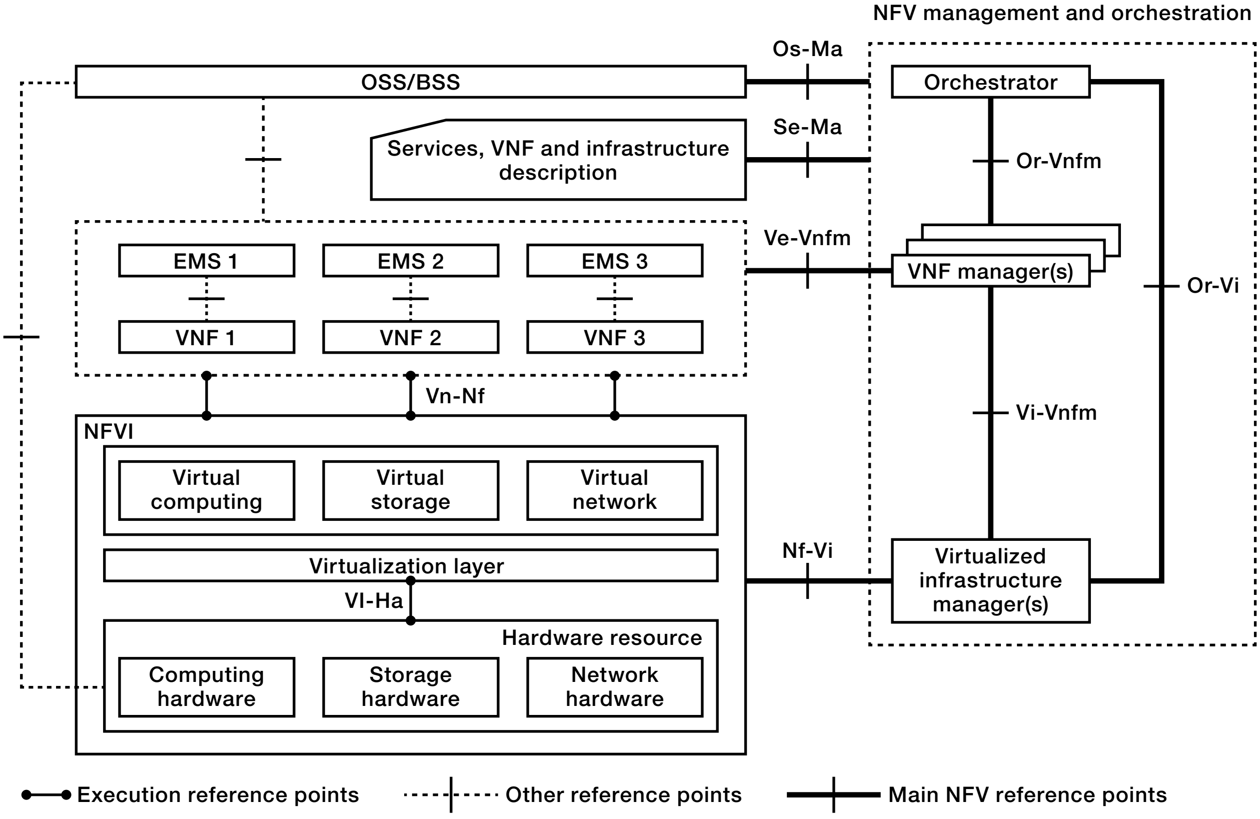


Figure 3-28 | ETSI NVF MANO framework architecture [source: ETSI NFV]

avoid timeouts by adapting itself to the current measured delays of response between the components.

The time to deploy new virtual machines/ containers currently is in the order of minutes. Using a non-intelligent system to deploy a new service instance available in this time range for services demanding ultra low latency might pose risks. Thus, the solution is to gain intelligence about the service usage itself and have the virtual machine resources pre-allocated in time, sometimes even having the virtual machines/ containers available "just in time" or from a pool of spare virtual machines, pre-provisioned with the service capabilities, so that they can be configured with ultra low delay and connected to the existing infrastructure topology.

3.4.5.4 Identity, authentication and authorization

In [42], some use cases where security plays a role are described, e.g. secure boot, secure crash, multiple administrator role, user/tenant authentication and authorization and accounting.

In the security specification of ETSI NFV [42], the virtualization software OpenStack is analyzed from the supported security point of view, e.g., authentication, authorization, confidentiality protection and integrity protection, and how this relates to ETSI NFV.

Among the many ETSI NFV MANO implementations, one has recently gained visibility by using multiple H2020 projects. The name of the toolkit is OpenBaton [43] and is capable of handling a set of data centres that run OpenStack or Amazon Cloud. In order to have a network topology deployed dynamically, one has to create an adaptor for each service to be running as VNF and then request the orchestrator to deploy the network service descriptor (NSD) on a specific point of presence (PoP), i.e. a data centre.

3.4.5.5 Multi-access edge computing

ETSI multi-access edge computing (MEC) working group provides a new approach to mobile core networking. Operators can open their radio access network (RAN) edge and place authorized third-party application functionality towards mobile subscribers, enterprises and vertical segments.

The mobile edge computing framework shows the general entities involved for enabling the implementation of mobile edge applications as software-only entities. These can be grouped into system level, host level and network level entities [44], see Figure 3-29. The mobile edge management comprises the mobile edge system level management and the mobile edge host level management.

The architecture of the MEC server is composed of middleware services to the applications [45]:

- Infrastructure services, including 1) communication services providing API to interact with the application services and between them, and 2) a service registry used by the applications to discover and locate the endpoints for the services they require.
- Radio network information services (RNIS) providing information to the applications to calculate and present the following

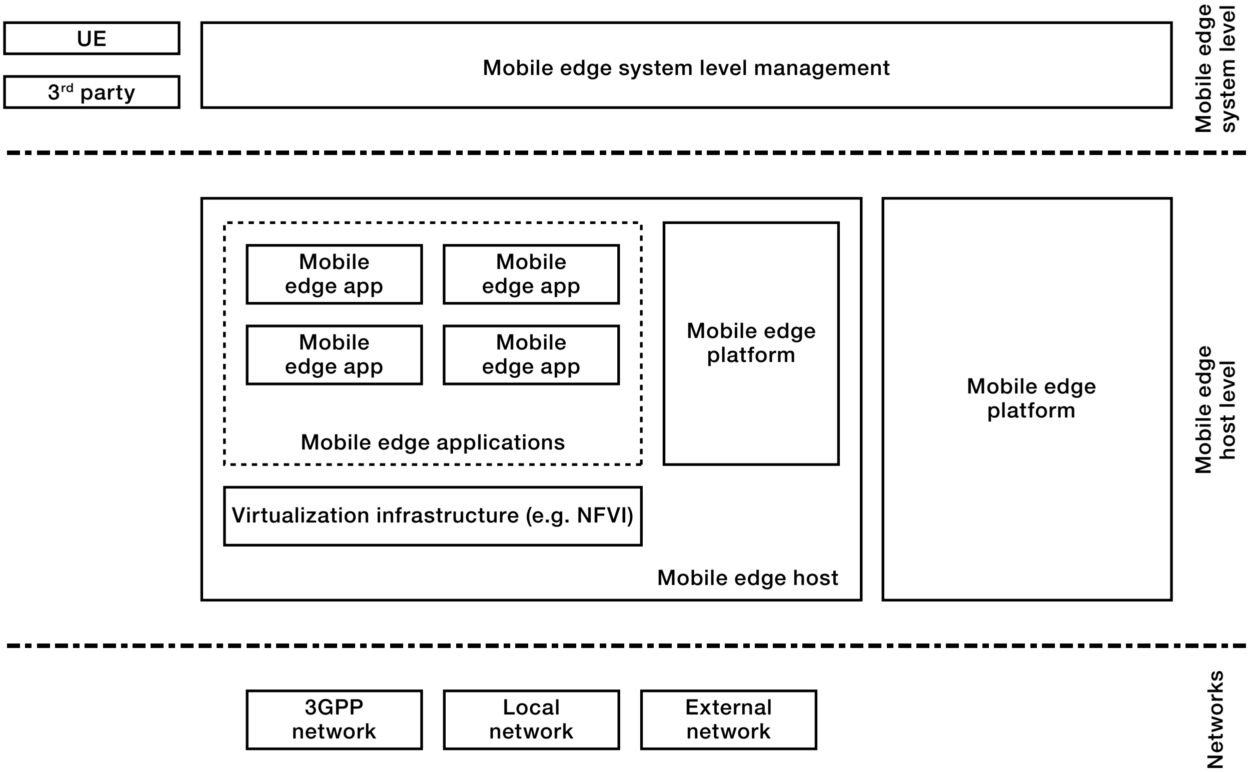


Figure 3-29 | ETSI MEC framework architecture [source: ETSI]

high-level and meaningful data: cell-ID, location of the subscriber, and cell load and throughput guidance.

- Traffic offload function (TOF) prioritizes traffic and routes the selected, policy-based, user-data stream to and from applications that are authorized to receive the data. It can act in a pass-through mode, in which the data plane is sent to an application and then to the original PDN-GW, or in an endpoint mode by serving the data to the location application.

In September 2016 the industry specification group (ISG) MEC changed the acronym MEC to "multi-access edge computing" [46] to reflect the importance of addressing Wi-Fi and fixed-line in addition to 3GPP access technologies. Opening up the radio access networks to third-party players can create value and opportunities for mobile operators and accelerate innovation for new services and applications that make use of proximity, context and speed available at the mobile edge.

3.4.5.6 Programmable connectivity identities

One of the limitations of the IoT infrastructures that are using mobile connections and are travelling or being transported or even loaned or sold to a new party is that the SIM identity can only be changed by manually changing the SIM card (universal integrated circuit card (UICC)). The solution for a secure remote provisioning architecture for the embedded UICC (eUICC/eSIM) was standardized by GSMA for both consumer and M2M SIM cards [47].

The eSIM is registered to a configured subscription manager and receives policies to activate/deactivate an operator connection and can download new credentials in order to connect to an operator, see Figure 3-30. The interface to handover between subscription managers is supported. The standard is very flexible and secure and enables after-market management of devices, making the lifecycle management of mobile or sold products very efficient.

Another solution for this problem is to embed the credentials in the firmware and send a firmware update with new credentials. An alternative is to store the credentials on a smart card and have a management client update the credentials based on the information received from an authorized entity, using OMA LWM2M device management protocol.

3.4.5.7 5G core network

Recently, 3GPP has started standardizing the core network architecture in [48], answering a list of requirements, many already presented in the NGMN 5G white paper [49]. The technical specification [48] introduces the architecture supporting LTE, NextGen Radio (5G) as well as WLAN and satellite.

In comparison to the 4G core network, the following outstanding new concepts regarding intelligence of the network are included in the NextGen architecture, see Figure 3-31:

- Network slicing and network slice selection: through this mechanism, leveraging NFV technologies, dedicated or isolated networks can be deployed on a shared hardware. This concept is very powerful and introduces a flexibility not present until now in the core networks. As a result, the core network transcends a transformation from a hardware-based to a software-based network, by preserving the interoperability using inherited protocols and standards, e.g. diameter for authentication, authorization and charging. Nonetheless, this concept, as well as the clear separation between the control plane components and user plane components, enables the necessary flexibility to introduce adaptation to scalability, security and mobility requirements.

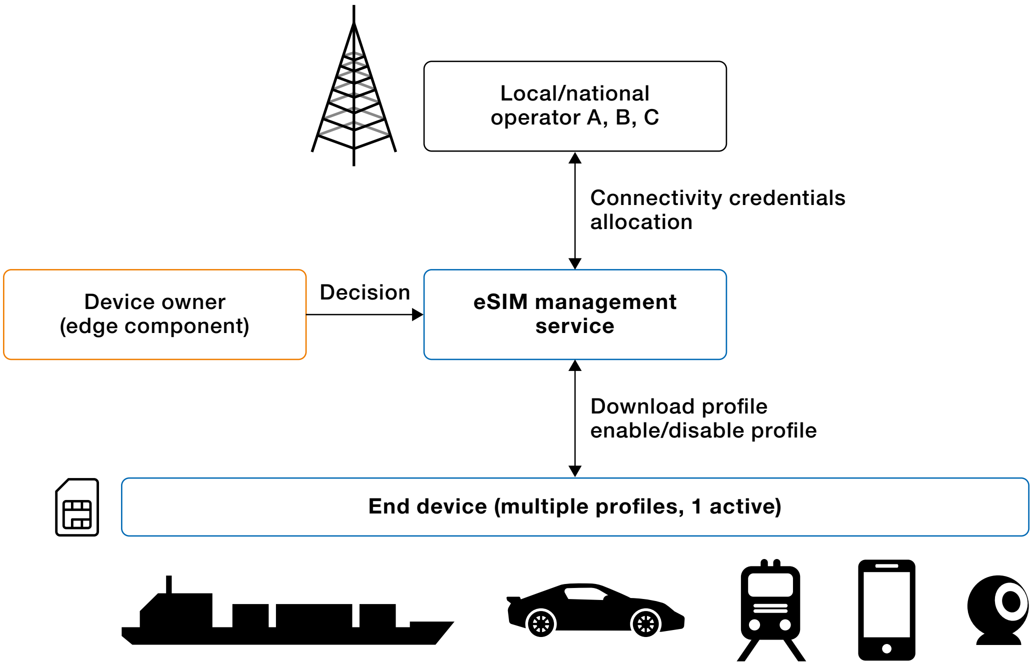

Local/national operator A, B, C
Connectivity credentials allocation
Device owner (edge component)
Decision
eSIM management service
Download profile enable/disable profile
End device (multiple profiles, 1 active)

Figure 3-30 | eSIM enabled infrastructure architecture

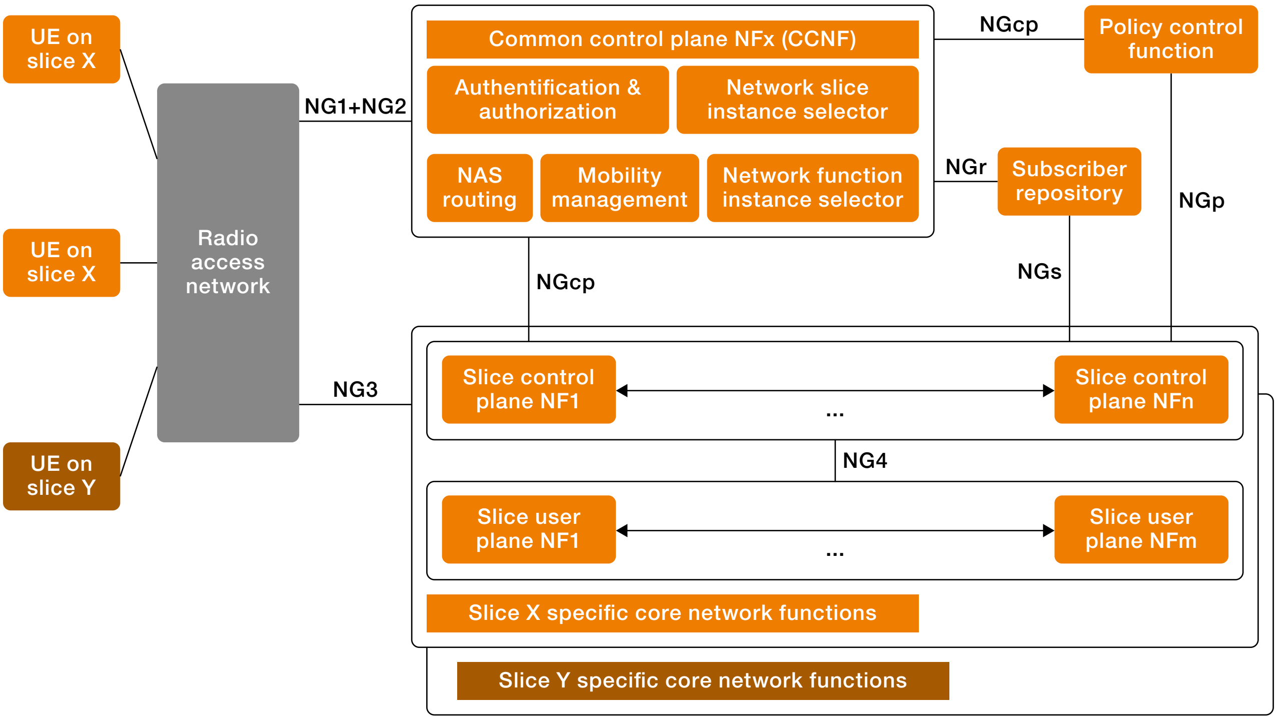


Figure 3-31 | 3GPP NextGen architecture

In the case that multiple dedicated networks are collocated, for example in a mall where a public network provides guests with non-critical QoS and another network with guaranteed QoS resources is deployed for serving public safety services or maintenance of the smart building infrastructure, the new mechanism called network slice selection, introduced by 5G, can be used in order to differentiate and allow access to one of the dedicated networks, see Figure 3-32.

- Scale-in, scale-out of components, software-defined networking design, energy efficiency and network capabilities exposure: these concepts together with network slicing are introducing flexibility and self-adaptation for energy efficiency and can be considered an important step toward having an intelligent network.
- Content-aware radio access network for QoS support: QoS capabilities are pushed right to the edge of the core network, in the software components running along with the base stations.
- Mobility framework supporting amongst others: mobility on demand concept and unreachability detection. The framework will be designed to be adaptive, flexible and intelligent to cater for disparate NextGen mobility requirements, e.g. energy efficient IoT communication that includes end devices that are classified according to their mobility needs: stationary, mobile in a certain area, highly mobile. The requirements of the devices will be taken into account in order to induce policies, and thus intelligence, in the network, in terms of determining the anchoring points (base stations, core network components).
- Service session continuity: even if a mobile device is moving, its application layer service session should not be interrupted or disconnected. The required capability is to determine the time to reselect a new user plane component to handle the application level

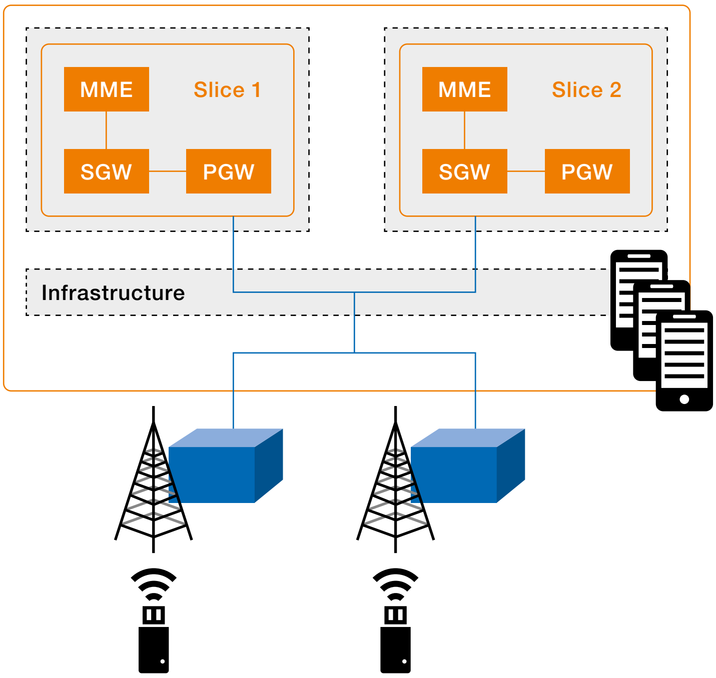


Figure 3-32 | Network slice selection

traffic, as well as determining the destination user plane component instance, having in mind that the service entity could reside close to the edge (radio access network).

3.4.6 Access networks

3.4.6.1 Narrowband IoT

Recently 3GPP has standardized the technology called narrowband IoT (NB-IoT) designed specifically for connecting the IoT [50]. The communication of the "things" is encapsulated as the payload of the non-access stratum (NAS) protocol via the MME, which was previously considered a pure control plane entity. Thus the "things" can send/receive data without establishing a default bearer. This new concept has enabled IoT infrastructures to have the battery life of sensors increased to several years.

With an extension of the EPC system, the 3GPP introduced handling of the NB-IoT messages.

This new architecture, see Figure 3-33, is called cellular IoT (C-IoT) and has the following advantages:

- Provides efficient support of infrequent small data transmission by minimizing network signalling.
- Eliminates the need for data bearer assignment by encapsulating user data payload into NAS signalling messages.
- Securely transports user data or SMS messages via the service capability exposure function (SCEF), designed especially for machine type communications (MTC) and delivery of non-IP data (NIDD) over the control plane.

3.4.6.2 Sigfox

Sigfox is a cellular network operator, which specializes in providing communication dedicated to M2M/IoT applications [51]. Communication is optimized towards low throughput within a radio cell with very large coverage. Typical data rates range from 10 bits per second up to 1 kbit per second; signal propagation in open space reaches 40 km. The employed ultra narrow band (UNB) technology operates in any license-free band, such as the 433 MHz, 868 MHz, or 2,4 GHz industrial, scientific and medical (ISM) bands, which allows for global deployment without the need to obtain a licensed spectrum [52]. Sigfox uses binary phase shift keying within a tiny piece of spectrum, which mitigates the effect of noise. To reduce the cost of devices, an asymmetrical link budget is typical. As such, Sigfox primarily targets data

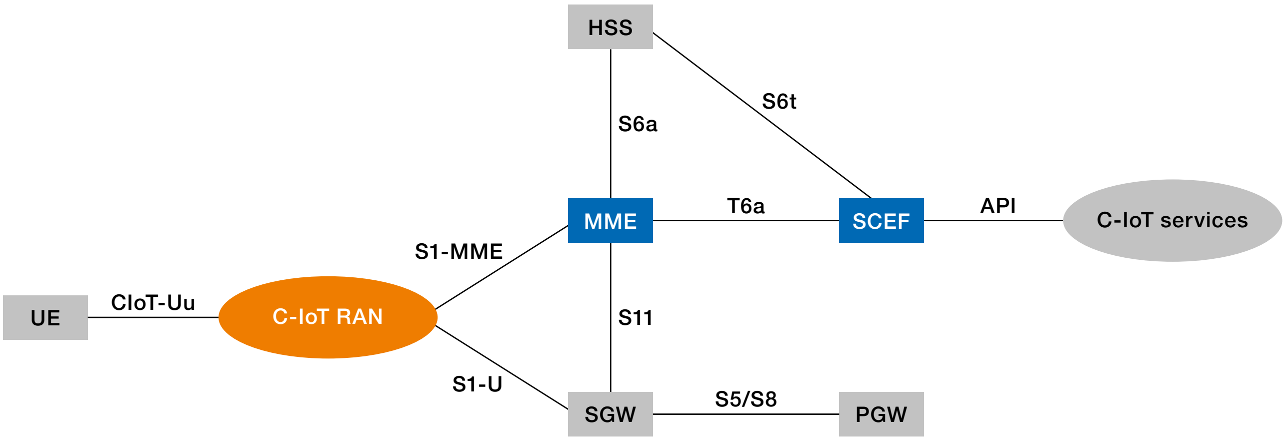


Figure 3-33 | Narrowband IoT control plane optimization for IoT

transmission from the device to the base station, which compensates for low signal-to-noise ratios by having expensive, sophisticated receiver technology, though downlink communication is also supported.

The limitation of using this network access is that all the data messages are sent three times in order to compensate for any loss, and service providers usually have one instance of the data server that receives and stores the data from all the devices. Towards users a simple HTTPS API is offered. Thus in total this gives an impression of limitations of employing intelligence in the network, e.g. privacy, identity management and access control, quality of service, as well as isolation of data streams that are related to different applications or use cases.

3.4.6.3 LoRa and LoRaWAN

LoRa and LoRaWAN are technologies specified by the LoRa Alliance. LoRa refers to the radio transmission technology applied, whereas LoRaWAN refers to the medium access control (MAC)-layer used on top of it.

LoRa is a spread-spectrum technology operating on wider bands, usually having a bandwidth of 125 kHz or more. In contrast to Sigfox, LoRa achieves increased receiver sensitivity via its frequency modulated chirp which results in a fully symmetric link budget similar to that of Sigfox but at an equally low cost for receivers at the device and base station sides [53].

LoRaWAN data rates range from 0,3 kbps to 50 kbps. Due to the employed spread spectrum technology, different rates do not interfere with each other. As such, LoRaWAN offers several virtual data channels in parallel, each having its own unique data rate [54].

From a network technology perspective, LoRa builds upon a star-deployment in which gateways are transparent forwarding bridges and in which the centralized master decides which gateway is best suited for relaying communication with a given end device, see Figure 3-34.

3.4.6.4 TSN

The control of autonomous systems is realized by successive control loops. A loop comprises three phases: data input, control computation, signal output. Each phase has deterministic timing to follow, and time-sensitive data must be transmitted within strict bounds of latency and reliability. In

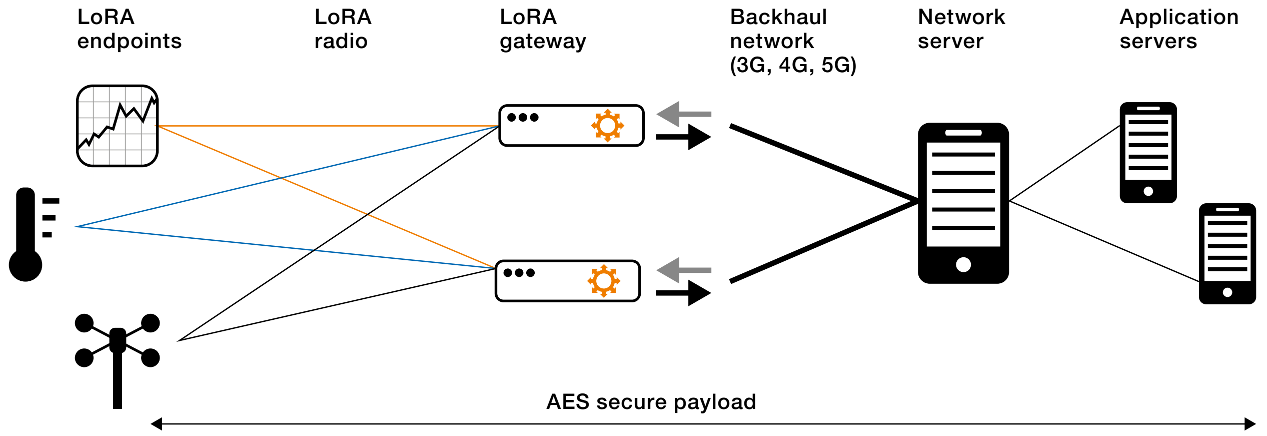


Figure 3-34 | LoRa functional architecture

the automotive and industrial domains, mission-critical and time-sensitive data, including sensor data and control signals, is usually transmitted over unidirectional dedicated cables, which leads to massive inflexible cabling. It would be better to converge all traffic classes of multiple applications (time sensitive or not) in one network, so that the operational expenditure (OpEx) can be reduced. An important candidate is Ethernet due to the corresponding mature components and technologies. However, the Standard Ethernet cannot be directly applied, because its best-effort mechanism cannot deal with time-critical traffic. Minimizing the average delay is the primary metric of best-effort mechanisms, whereas the deterministics of traffic is not guaranteed.

Time-sensitive networking (TSN) is a set of Standards under development by the TSN task group of the IEEE 802.1 working group. TSN, which is built on the MAC/PHY layer of Ethernet, offers a low latency, time deterministic and highly reliable way to send time-critical traffic over standard Ethernet infrastructures. The three important features of TSN are time synchronization, scheduling and time-aware traffic shaping. The traffic shaping controls logical gates on the network switches according to the schedule, so that specific time windows are reserved for time-sensitive traffic, whereas the other traffic is blocked. Accurate time synchronization is the prerequisite for scheduling and shaping.

In case an ECN is required to interface with devices in industrial automation, automotive or robotic environments over Ethernet, support for TSN may be needed to prioritize time-sensitive traffic in crowded networks.

3.4.6.5 WIA-PA

Pervasive sensing and field communication technologies are basic to realizing smart manufacturing.

While numerous challenges exist in some industries, for example industrial automation, the control application cycle time should be within 2 ms-50 ms, the transmission delay should be less than 10 ms, and the packet loss rate should be less than 0,01%. To the wireless communication technologies, the index of reliability/latency is more scale sensitive due to co-channel interference (CCI). As the number of wireless mesh nodes increase, the latency index has a nonlinear growth, and the reliability index has a nonlinear reduction. The wireless networks for industrial automation process automation (WIA-PA) brings forward some innovative technologies such as adaptive frequency to address these challenges, see Figure 3-35. WIA-PA became a national Standard (GB/T 26790.1-2011) in China in July 2011, and became an IEC International Standard (IEC 62601) in October 2011.

3.4.7 Programmable infrastructures

Recently the trend is to softwareize the network, the service components, even the time and location where new instances of such components have to be deployed.

At the same time the end devices, especially the ECNs, are also leveraging the cost efficiency of running powerful hardware and intelligent software in order to solve complex problems.

The need for communication between the multiple ECNs or between ECNs and core components remains. For most infrastructures requesting dynamic growth and scalability, the network and service functions can be deployed dynamically to cope with peak times and shut down to improve energy efficiency.

3.4.7.1 Programmable communication manager

Using old concepts such as load-balancer might introduce a bottleneck and increase delay in communication. The next step in the evolution of the infrastructures that contain both ECNs and

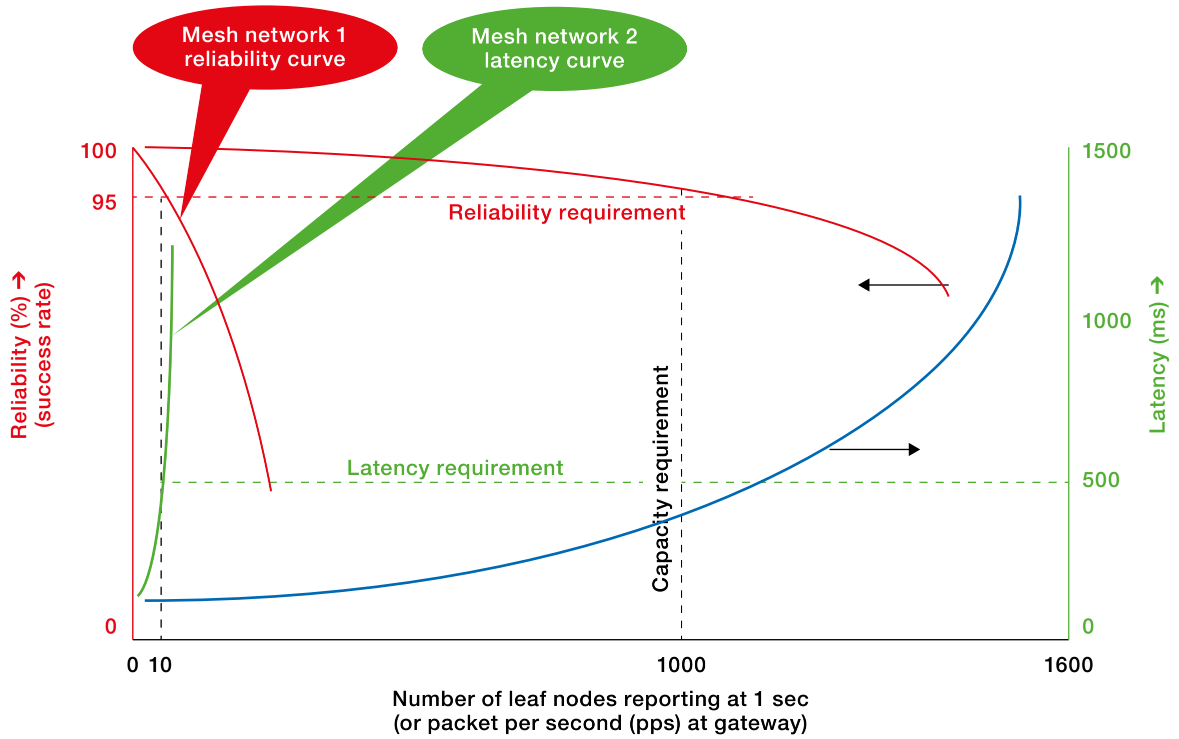


Figure 3-35 | WIA-PA source

core components is to have the ECNs adapt to infrastructure changes – a tremendous advance.

At the same time, solutions for directly coupling the ECNs to their serving network function involve from the outset removing the delay of the load-balancing algorithm and enabling more complex logic in terms of organizing the network by adapting to regional policies or ECN capabilities and constraints in terms of delay or routing.

Such communication solutions can use the intelligence of the communication manager regarding the ECN capabilities and constraints to have as input the topology changes advertised by the NFV orchestrator in order to push to the ECNs the recommended communication peer (e.g. a service instance), as can be seen in Figure 3-36. Such a mechanism making use of device management protocols is already implemented in the Open5GMTC toolkit [55]. This intelligent device communication manager component can even reside on the base station and manage the IoT devices in its range, in order to minimize reconfiguration delay.

3.4.7.2 Local breakout

Local breakout is a mechanism from which the eNodeBs can forward the traffic directly to an application server and not through the internet gateway outside the core network. The Standard [56] does not specify the case in which the end device is mobile, i.e. is performing a handover between two eNodeBs.

In the case of a moving end device which requires service continuity, e.g. in an automotive or logistic use case, a solution for local breakout from the base station towards the server instance located near the ECN is necessary. A prototype implementation is available in the Open5GCore toolkit [38], having the handover delay between two software eNodeBs as a third of the delay of the multimedia service, compared to the case when the service instance would be directly connected to the outside network breakout, see Figure 3-37. This confirms the fact that the intelligence on the base-station to route the multimedia packets, to be used for example in content delivery networks, can bring benefits in terms of quality of service and quality of experience, even while on the move.

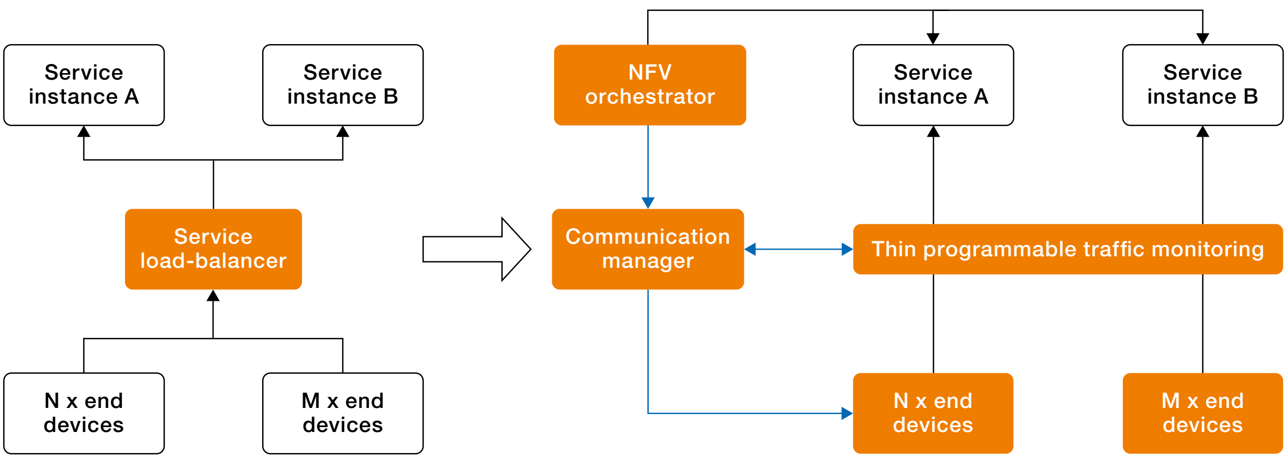


Figure 3-36 | Towards network topology-aware infrastructures

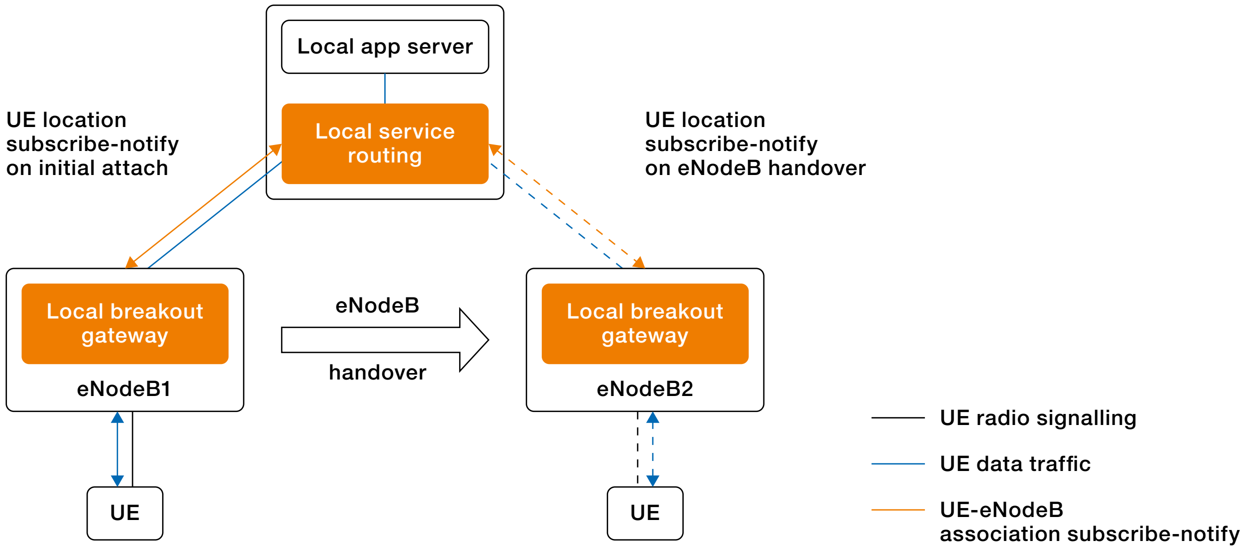


Figure 3-37 | Local breakout application enablement

Section 4

Use cases and requirements for edge intelligence

This section introduces a series of use cases that will need to be available to support the informational transformation toward the application of anticipated EI technologies within different industry segments. The use cases covered and the requirements identified relate to this process of transformation rather than to the eventual application of the EI technologies themselves.

A total of seven use cases are referenced, covering the following specific themes:

- Factory productivity improvement
- Connected city lighting
- Smart elevators
- Indoor location tracking
- Lone worker safety
- Access control – tailgating detection
- Fire detection via surveillance cameras

Section 4.1 provides a short description of the various use cases, with more detailed descriptions of these included in Annex A. Each detailed description is comprised of 1) the scope and objectives, 2) use case diagrams, and 3) a technology assessment of the state of the art, remaining technical gaps, needed capabilities and necessary future Standards.

Section 4.2 presents a structural overview of the use cases in the form of a framework for discussion, with the intent of setting the floor for other potential EI use cases.

4.1 Use cases overview

- **Factory productivity improvement**

For factory operators, it is important to improve factory productivity, so that the profit generated by the factory can be maximized. Big data analysis realized through a combination of edge computing and cloud computing can contribute substantially to factory productivity improvement. Edge computing with intelligence converts data collected from the shop floor into data which can be analyzed by cloud computing. The analysis result leads to minimization of products determined to be defective by mistake and also enables predictive maintenance, which can reduce factory down time.

- **Connected city lighting**

While bringing convenience to people's lives and decorating urban areas, city lights consume a vast amount of energy and increase management costs for municipalities. To address these issues, the solution of connected city lighting is being developed. Firstly, the solution provides integrated smart lighting policies to ensure high energy efficiency, in which EI helps control the lights according to time, brightness, weather and many other features. Secondly, online inspection makes it possible to monitor the status of each light in real-time, with the maintenance staff being informed automatically once a malfunction occurs. The traditional manual inspection is no longer necessary, therefore OpEx can be greatly reduced. Thirdly, the local survival mechanism uses the local cached policy to control the lights when the remote control in the cloud is unreachable.

In the future, light poles will move from performing a single function (the lighting) to being multi-functional, i.e. many other functional modules can be added on, such as environment/utility monitoring, video surveillance, vehicle-to-infrastructure (V2I) communication devices, etc., making the light poles become an integrated system of sensing and service provision.

- **Smart elevators**

Urbanization makes elevators indispensable in cities. The operation and maintenance of elevators is considerably expensive due to manual inspection, time-consuming fault detection and repairing. Smart elevator solutions with edge and cloud intelligence can help vendors upgrade from the inefficient, expensive preventive maintenance model to next-generation, real-time, targeted, predictive maintenance, extending value from products to services.

Hundreds of sensors are deployed to monitor the elevator's status. Based on this data, the ECN is capable of detecting potential device faults early and sending out the alarms immediately. When the ECN fails to connect to the cloud, the data can be stored locally until the connection recovers. By analyzing the historical data at the cloud, faults can even be predicted, so that maintenance is given accordingly before a fault actually occurs. New features of faults can also be extracted by AI at the cloud and then downloaded by the ECN.

- **Indoor location tracking**

The bandwidth requirements for indoor location tracking, are moderate: approx. 2 MB, with very low latency (<1 ms) and low contention. The system requires a backhaul of trilateration data for a number of sensor sources (all normalized to IP/user datagram protocol (UDP) packets) and conversion into a high quality location estimate. As it is used for high value asset tracking, latency is a real problem, so having a gateway which processes the sensor samples as close as possible to the source, while still maintaining the gateway in the cloud, or at least outside the customer premises, is critical for a system of this type. Within the gateway module is an embedded, tunable, MI module to perform the location estimation, which then forwards real world positions and user status to the administrative/UI module in the cloud. The model may also be tuned and the MI module updated.

- **Lone worker safety**

For lone worker safety an intelligent gateway is used to receive signals indicating the location of a particular employee. In such cases an MI module would also be embedded in the GPS signal transmitter, which would use an MI module to characterize the wearer's gait and orientation. The module would "learn" over a period the "normal" behaviour of an individual, and thus be able to generate an alarm should that behaviour change due to accident, attack etc. The transmission bandwidth would be very low and latency could be in seconds, as decision-making would be local. The intelligent gateway in this case would function only as a data consolidator and could also have UI, etc. installed.

- **Access control – tailgating detection**

Using an MI module attached to an inexpensive stereo camera, the system performs head and shoulders detection, based on a starter dataset, to eliminate tailgating at security doors. Coupled with this, an additional reader Bluetooth low energy (BLE)/RFID for example) enables a badge to be read and thus allows frictionless access through the door, i.e. the door opens as the person approaches, but only if a valid signal is read from each user (if there is more than one) approaching. If more than one person is detected approaching, and fewer badge signals are read, then if those people attempt to go through the door an alarm will be registered. The inexpensive camera is linked to a processing module, located either in-premise or in the cloud. Latency and bandwidth are both issues in this case.

- **Fire detection via surveillance cameras**

An MI module is used with access to video streams incoming from non-specialized video cameras and scans those streams for traces of a fire using a trained model. Cameras are linked to a processing module, located either in-premise or in the cloud. Latency and bandwidth are both issues in this case.

4.2 Framework for discussion

Table 4-1 summarizes the seven use cases together with the associated identified technology gaps and needed capabilities. The table serves as a framework for discussion of other EI use cases from the same industry domain or from other domains that might face similar technology gaps, so that they too can find here the needed capabilities to overcome such gaps. One can also conclude that some of the needed capabilities serve multiple use cases. For example, ML, self-organization by preserving the security of identity, containerization and self-configuration, are present in more than one use case.

Sections 5 and 6, describing the technology gaps and resulting needed capabilities, base their observations and propositions on the framework provided in Table 4-1.

Table 4-1 | Identified use case technology gaps and needed capabilities

	Technology gaps	Needed capabilities
Factory productivity improvement	▪ Credibility of information generated by edge computing ▪ Assisted/automatic optimization of system operation ▪ Edge computing environment which allows applications to be deployed	▪ Management of secure identity – as an enabling technology for credibility of information generated by edge computing – e.g. physically unclonable function (PUF) – to ensure that information is not hampered by unauthorized parties during the communication ▪ ML technology – as an enabling technology for automatic optimization of system operation
Connected city lighting	▪ Convenient function add/remove ▪ Evolutionary light policies and energy management policies ▪ Cloud offloading and privacy: processing the data locally at the edge ▪ Interaction between ECNs, context synchronization, control and orchestration	▪ Self-organization, self-discovery, plug and play ▪ ML for ECNs to support learning abilities ▪ Processing power, storage and networking, security policies for ECNs ▪ ECNs support E/W-bound communication

	Technology gaps	Needed capabilities
Smart elevator	▪ Diagnosis and predictive maintenance ▪ Cloud offloading and privacy ▪ Interaction between ECNs ▪ Security in preventive maintenance	▪ ML for ECNs to support learning abilities ▪ Processing power, storage and networking, security policies for ECNs ▪ ECNs support E/W-bound communication ▪ Decentralized authentication mechanisms like blockchain
Indoor location tracking	▪ Bandwidth is not often available at the levels required for accurate tracking ▪ Level of calculation required for accurate tracking requires high level of compute	▪ Tracking nodes once installed require considerable bandwidth, and E/W communication between tracking nodes will allow lower N/S bandwidth requirements ▪ Higher local processing capacity in smaller packages, ideally included as a container in another node, in either camera or other edge node ▪ Containerization and self-configuration are required to be implemented at a higher level to allow easy, trusted installation
Lone worker safety	Some packages exist, but have not been adopted widely, such as: ▪ footprint is bulky ▪ false alarms are frequent, due to the system interpreting a normal movement, e.g. a fall ▪ backhaul connectivity is often lost ▪ loss of connectivity countermeasure	▪ Narrowband IoT protocols such as Sigfox to reduce signal loss, coupled with deployability on a worldwide basis ▪ Better battery performance for GPS operation, or lower power GPS operation. This may also be implemented by intelligent operation of the device to extend battery lifetime ▪ Greater local compute, or specialized MI capability on embedded processors to allow MI algorithms to reduce false alarms ▪ End device policy concerning cost, accuracy or other criteria for switching between connectivity options ▪ Pre-provisioned credentials for connecting to the optional networks

	Technology gaps	Needed capabilities
Access control – tailgating detection	▪ While this can be achieved currently, both cameras and compute nodes are not cost effective for the solution ▪ Solution is challenging to install due to calibration to an environment.	▪ Cost reductions in stereo camera modules ▪ Greater local compute, or specialized MI capability on embedded processors to allow MI algorithms ▪ Containerization and self-configuration are required to be implemented at a higher level to allow easy installation
Fire detection via surveillance cameras	▪ While this can be achieved currently, both cameras and compute nodes are not cost effective for the solution ▪ Solution is challenging to install due to calibration to an environment	▪ Cost reductions in stereo camera modules ▪ Greater local compute, or specialized MI capability on embedded processors to allow MI algorithms ▪ Containerization and self-configuration are required to be implemented at a higher level to allow easy installation

Section 5
Technology gaps

This section describes the different EI gaps that need to be addressed by further development to better support the use cases and meet their requirements.

5.1 Factory productivity improvement

- **Credibility of information generated by edge computing**

In order to realize EI, it is necessary to establish the credibility of the data which constitute the inputs for analysis and decision-making. At present, certain technologies can be used as a component technology to secure the credibility of data exit, such as technologies to secure the identity and the integrity of device-generating data and technologies to protect information being communicated over a network. In addition to enhancing those component technologies, it is desirable to establish systematic ways to guarantee the credibility of data as a system by making use of such technologies and operational measures including physical security.

- **Assisted/automatic optimization of system operation**

In order to reduce the operational cost of a system, it is necessary to optimize the way the system is operated or adjust operational parameters more efficiently. At present, decisions about how to improve system operations such as the productivity of a factory are done manually by experienced engineers, and even the acquisition of data necessary for such decision-making is done manually in some factories. It would be desirable to make optimization easier by first assisting engineers in the analysis of data for optimization and ultimately by automating optimization processes using the know-how of expert engineers made available by MI in some areas.

- **Edge computing environment which allows applications to be deployed**

At present, there are multiple vendors of edge computing products, and the edge computing (programme execution) environment of those products is proprietary. This situation makes it difficult for developers of edge computing applications to develop a portable application which can run across multiple edge computing products. It would be desirable to utilize the wisdom of people from various disciplines in system development by enlarging the developer community, as is the case with smartphone applications. Making the edge computing environment open can contribute to that purpose. As for applications for infrastructures such as those in manufacturing, it can be necessary to consider how to guarantee that an application does not behave maliciously and is not harmful to a system, by some measure such as certification of an application. Lifecycle management of an application is also an issue to be addressed.

5.2 Connected city lighting

- **Pole networking, function add/remove**

When deploying lighting poles, the installed function modules should first be identified, and then neighbouring devices and ECNs specified with which to establish connections. In this way, the network can be built without much human intervention. The networking topology should be flexible according to certain conditions such as connection qualities and the computation load of ECNs. Once a function module is added or removed from a pole, the network and the applications at the ECN should change accordingly.

- **Evolutionary lighting policies and energy management policies**

The basic lighting policy is drawn up based on the time, date and geographical location. Some reactive policies can be made according to the environmental brightness and the proximity of vehicles and pedestrians. The edge and the cloud should have learning abilities to build continuously evolving rules. Since lamps and the other modules, especially the charging module, are part of the smart grid, the energy management policies should also be optimized.

- **Cloud offloading and privacy: process the data locally at the edge**

The allocation of processes between the edge and the cloud must be defined: the time-critical data must be processed at the edge to get a timely response; some lower stage processing can be conducted at the edge, such as filtering and aggregation, so that the cloud can be offloaded; for privacy reasons, some data must be processed locally, e.g. in video surveillance, only abnormal events are reported to the cloud while the citizens' portrait should be protected.

- **Interaction between ECNs, user information synchronization, control and orchestration**

In some scenarios, the ECNs need to work in a cooperative way. For instance, at night, lamps need to increase their brightness when vehicles and pedestrians approach and decrease the brightness when they leave. This process is expected to be continuous, especially the handover between two ECNs. Based on the headings, velocities and positions, an ECN can predict the next ECN that the vehicle will pass and remind the latter to get ready for handover.

5.3 Smart elevator

- **Diagnosis and predictive maintenance**

Since elevator safety is strongly related to public safety, the fault detection and maintenance is considerably important. Currently, maintenance on site requires rich experience in diagnosis and repairing, which leads to extra expense in staff training. Moreover, it often takes a considerable amount of time to identify the fault and find the corresponding solution.

Autonomous, accurate and timely diagnosis helps locate the malfunction once it happens, while predictive maintenance gives the alert before malfunctions occurs. Both of these mechanisms will help to improve the safety of the elevator and reduce the OpEx of companies.

- **Cloud offloading and privacy**

The missions of the edge and the cloud should be well separated: functions that demand a timely response, such as fault detection, should be realized at the edge; the edge can also handle data pre-treatment such as filtering and aggregation, only critical data being uploaded to the cloud; preventive maintenance could be conducted at the cloud by comparing the uploaded data and the labelled samples. It should also be considered, for example, that the data and video surveillance data of manufacturers needs to be processed locally, according to privacy needs.

- **Interaction between ECNs**

Smart elevators may also demand cooperation between ECNs. In large buildings, there may be several groups of elevators. When a group is fully occupied, by information-sharing between ECNs, passengers can be guided to the nearest available elevators to save time.

- **Security in preventive maintenance**

Malfunction reporting and data uploading should be authenticated and encrypted. On-site engineers must also be authenticated, then authorized to access the maintenance and repairing.

5.4 Indoor location tracking

- **Network bandwidth**

High local network bandwidth (with reasonably low latency <100 MS) and provisions for backhaul to a local processing node with sufficient compute to allow the calculations necessary will be needed.

- **Provisioning**

The ability to add this capacity seamlessly within existing and future networks should be provided.

- **Edge compute power**

The local processing node should be capable of a high degree of localized processing, at the level of what would be (in 2017) a server class machine. Ideally the processing logic should be containerized, or modularized in such a way as to operate seamlessly in spare space on general IT equipment or, if developments in embedded hardware allow, on a gateway scale compute node.

5.5 Lone worker safety

- **Power efficiency**

Current equipment is physically quite bulky due to the power/antenna requirements of current systems as well as the power requirements for maintaining GPS operation.

- **Machine intelligence**

In addition false alarms are frequent as the processing power onboard is not sufficient to allow MI to tell the difference between normal movement, or work activity, and an exceptional situation such as a fall, nor is the processing sufficient to allow intelligent power management of the peripheral devices.

- **Data transport cost**

The cost of current transport mechanisms is not justified by the level of data being transported.

5.6 Access control – tailgating detection/fire detection via surveillance cameras

- **Machine intelligence/strong edge compute**

Greater local compute is required, or specialized MI capability on embedded processors, to allow MI algorithms to be implemented on embedded devices/gateways.

- **Containerization**

The ability to be installed seamlessly and with a high degree of modularity and security in existing environments is essential. In nearly all cases such applications will be add-ons to existing camera systems and fire/security infrastructures.

Section 6

Needed capabilities

This section describes in more detail the needed capabilities that address each of the described gaps.

6.1 Integration of edge and core

In order to ensure data privacy and prohibit any data or system tampering, IoT edge computing solutions are expected to be securely integrated with the core. Edge solutions also need to be managed to minimize costs and to optimize lifecycle management across a wide range of edge devices. Data management and processing can take place at the core or edge, whichever approach is optimal for the specific scenario.

6.1.1 Edge services

Needed edge services under development include the following.

- Persistence service: to store IoT data on IoT gateways. IoT administrators can configure which data should be stored locally and set a data aging policy.
- Streaming service: to analyze IoT data streams. IoT administrators can define conditions with adjustable time windows to identify patterns in the incoming IoT data as a basis for automated events. For example, certain conditions can initiate transactions and notification of appropriate parties.
- Business transaction service: to execute business transactions at the edge to provide continuity for critical business functions, even when the edge is disconnected from the core.
- Predictive analytics service: to use predictive models for analyzing the IoT data. The predictive algorithm is constantly "being trained" and improved in the core based on all available data. The resulting predictive model is then sent to the edge and applied there.
- Machine learning service: to apply ML algorithms at the edge specifically for image and video analysis.
- Visual analytics service: to explore visually IoT data stored on IoT gateways. IoT data analysts can visually inspect the data collected at the edge. For example, after an alert has been sent to the core, an analyst can dig into the details which led to the alert.
- 3rd party application hosting service: to allow 3rd party application containers to be run on edge hardware, allowing decoupling between hardware and applications. For example an edge gateway might be used to run several services such as camera, access control, AC management, elevators.
- End-to-end sophisticated management system: to apply software-defined networking and other paradigms from 5G and other sources, in order to enable new business models on the EI domain based on tightly integrated services and networking.

6.2 Factory improvement productivity

- **Management of secure identity**

  It is necessary to validate that devices including ECNs are the intended ones in order to guarantee the credibility of data generated by those devices. A secure identity that cannot be duplicated easily and its management are useful for the validation. Secure identity can be used to identify a thing. It can also prevent malware from updating firmware by protecting the firmware with a shared secret key which is encrypted by the secure identity to be stored in the device. Example technologies for realizing secure identity are the PUF and the trusted platform module (TPM).

- **Communication security (confidentiality, integrity, authentication)**

  For realizing confidentiality, integrity and authentication in communication, several technologies already exist, such as transport layer security (TLS) and datagram transport layer security (DTLS) for IP communication. DTLS is a TLS for resource-constrained devices. TLS and DTLS use a digital certificate or shared secret key. Operations, administration and maintenance (OAM) of digital certificates or shared secret keys in the IoT edge computing system scale needs further investigation to determine whether the current OAM method for IT systems is sufficient. Some field networks which connect resource-constrained devices to edge computing systems use non-IP communication technologies. The treatment of confidentiality, integrity and authentication in those field networks needs further consideration, e.g. depending on the system configuration, the absence of confidentiality, integrity and authentication measures in those field networks is acceptable and an ECN acting as a gateway to the rest of the system provides some protection for them and the rest of the system.

- **Machine learning technology**

  ML technology is effective for optimizing the system operation according to changes to the environment in which the system is operating. There are two implementation patterns. The first involves having the cloud learn models and distribute these learned models to the ECNs and having the ECNs execute the model and analyze or make decisions. The second involves having the ECNs learn models directly and execute those models. It is desirable to develop ML algorithms which can be applied to problems that offer few learning samples due to the rare occurrence of target events, e.g. failure prediction.

6.3 Connected city lighting

- Self-organization, self-discovery, plug and play: by horizontal decoupling, the different function modules can be connected to the ECN with standardized lifecycle management interfaces, and the corresponding applications can run on a common open platform, even though the function modules belong to different vendors. When a module is added or removed, the application is added/removed accordingly. In this way, the modules can be managed with agility and repetitive development is avoided.
- AI for ECNs to support learning abilities: different ML algorithms can be deployed in some applications such as smart lighting policy, vehicle charging and video surveillance to provide optimized services. During operation, AI is given new samples so that the intelligence evolves continuously.
- In order to offload computing from the cloud, the ECNs must have processing, storage and networking abilities. This means sufficiently powerful CPUs and adequately sized RAM and FLASH to support these abilities. Since private data is stored at the ECN, security policies should be implemented to prevent attacks from outside.
- E/W-bound communication should be enabled to realize the interactions between ECNs.

The data ontology between different kinds of ECNs should be coordinated to enhance data interoperability.

6.4 Smart elevator

- AR/VR technologies: based on image recognition and the diagnostic results given by the smart elevator system, AR/VR devices can help the engineer to locate the fault and propose a solution to fix it. If the fault is not in the database, the engineer can demand remote support provided by human experts or AI. In this way, the training costs of staff and the travel costs of experts can be saved.
- AI: the data currently acquired and the related maintenance operations (labels) can serve as the training samples in supervised learning. The training can be conducted at the cloud and the outcoming strategies can be embedded at the edge, so that the edge can respond in a timely fashion. The strategies can evolve when new samples are added in. The learning can also be unsupervised, with the AI tapping into the tremendous amount of data available to discover the implicit relations behind the data, thus enabling new strategies to be built spontaneously for predictive maintenance, even those not discovered by engineers.
- The ECN, in this case the gateway, should have processing power, storage and networking abilities supported by corresponding hardware. By horizontal decoupling, the ECN can communicate with elevators of different manufacturers, and third party applications can be deployed on the common OS. The private data is stored and encrypted locally.
- E/W-bound communication should be enabled on ECNs to support task coordination among them. Data ontology consistency must be guaranteed to enable interoperability between ECNs of different manufacturers.
- Decentralized authentication mechanisms such as blockchain: authentication of maintenance, including on site and remote, is necessary to prevent tampering or revision. If the authentication is only conducted by the cloud, once the control centre is tampered with, it becomes possible to control all the elevators. Decentralized authentication mechanisms such as blockchain are helpful to solve this problem. A block contains a timestamp and a link to the previous block. None of the modifications of the data can be altered retroactively. Even if an ECN is tampered with, all the other nodes will be notified and the messages sent by this node will be considered to be untrustworthy, so that it is impossible to attack the entire network from a single node.

6.5 Indoor location tracking

- **Bandwidth**

  Bandwidth increases within both public and private radio networks (see 5GPP) as well as the capability to hand back and forth seamlessly between these networks will allow tracking networks of this type to be fully realized.
- **Edge compute**

  Increases in embedded compute capacity are necessary to support systems of this type, with localized MI capability (at least at the level of feature comparison) also being useful.
- **Containerization**

  Containerization and self-configuration need to be implemented at a higher level to allow easy installation. Containerization will allow software to be delivered in discrete, secure, bounded packages, in which the API connections can be tightly defined via microservices, without any attendant library or development language/OS requirements. Containerization will allow multiple IoT applications from different vendors to coexist on processing hardware and to be maintained and upgraded separately with no reference to applications of other vendors or effect on their operations, while allowing information-sharing along pre-defined E/W interfaces at the device or gateway level.

6.6 Lone worker safety

- **Narrowband IoT**

  Narrowband IoT protocols such as Sigfox to reduce signal loss, coupled with deployability on a worldwide basis.

- **Machine intelligence/battery capacity**

  Better battery performance for GPS operation, or lower power GPS operation. This may also be implemented by intelligent operation of the device to extend battery lifetime.

- **Edge compute**

  Increases in embedded compute capacity at lower power are necessary to support systems of this type, with localized MI capability (at least at the level of feature comparison) also being useful.

6.7 Access control – tailgating detection/fire detection via surveillance cameras

- **Edge compute**

  Increases in embedded compute capacity at lower power are necessary to support systems of this type, with localized MI capability (at least at the level of feature comparison) also being useful for both supporting the recognition algorithm and allowing operation with more basic camera equipment.

- **Containerization**

  Containerization and self-configuration need to be implemented at a higher level to allow easy installation. Containerization will allow software to be delivered in discrete, secure, bounded packages, where the API connections can be tightly defined via microservices, without any attendant library or development language/ OS requirements. Containerization will allow multiple IoT applications from different vendors to coexist on processing hardware and to be maintained and upgraded separately with no reference to the applications of other vendors or effect on their operation, while allowing information-sharing along pre-defined E/W interfaces at the device or gateway level.

- **Network bandwidth**

  Bandwidth increases within both public and private radio networks (see 5GPP) as well as capability to hand back and forth seamlessly between these networks will assist in decoupling cameras and processing nodes.

Section 7

Standards and role of open source

This section describes the standardization/open source activities required to support the previously identified needed services and capabilities. As can be seen in Figure 1-2 (repeated here as Figure 7-1), some of the needed Standards apply to multiple use cases and application types, for example credibility/trust/decentralized trust, implementation of ML as well as self-organization and self-discovery.

7.1 Standards for self-organization, self-configuration, self-discovery

There is no doubt that with the growth in the number of ECNs, the management of the network, the ECN and the application will become a huge challenge. To facilitate the deployment of ECNs, it is better to mask the complexity of the technology from operators and users, and to realize the plug and play of devices. Therefore it is necessary to introduce autonomic networking. Currently, the autonomous functions already exist. However, the discovery, node identification, negotiation, transport, messaging and security mechanisms, as well as non-autonomic management interfaces, are being realized separately. This isolation of functions is leading to high OpEx.

The Autonomic Networking Integrated Model and Approach (ANIMA) working group of the Internet

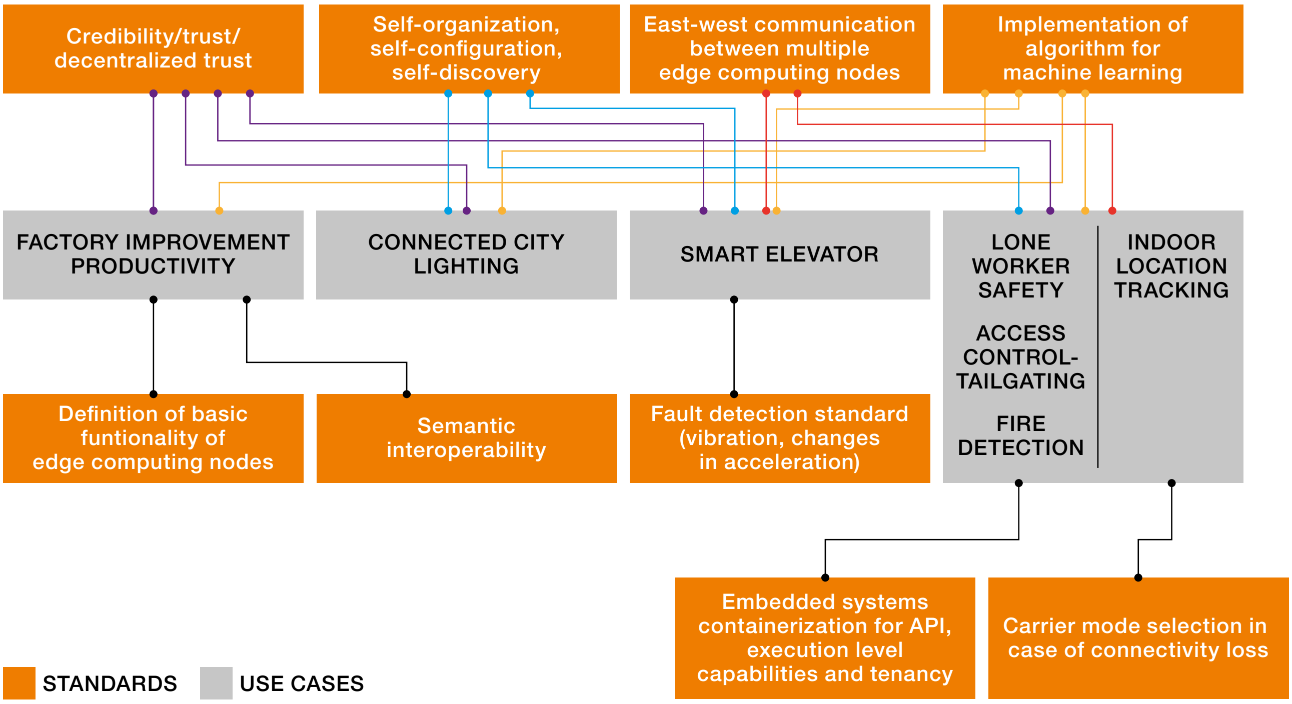

Credibility/trust/ decentralized trust
Self-organization, self-configuration, self-discovery
East-west communication between multiple edge computing nodes
Implementation of algorithm for machine learning
FACTORY IMPROVEMENT PRODUCTIVITY
CONNECTED CITY LIGHTING
SMART ELEVATOR
LONE WORKER SAFETY
ACCESS CONTROL-TAILGATING
FIRE DETECTION
INDOOR LOCATION TRACKING
Definition of basic funtionality of edge computing nodes
Semantic interoperability
Fault detection standard (vibration, changes in acceleration)
Embedded systems containerization for API, execution level capabilities and tenancy
Carrier mode selection in case of connectivity loss
STANDARDS
USE CASES

Figure 7-1 | Standards needed for EI

NOTE This figure is identical to Figure 1-2 and is repeated here for the convenience of the reader.

Engineering Task Force (IETF) is developing a system of autonomic functions to manage the network at a higher level without detailed low-level management of individual devices. In a secure closed-loop interaction mechanism, the network elements cooperate to satisfy management intent: network processes coordinate their decisions and translate them into local actions.

In the cellular communication domain, 3GPP has proposed the self-organizing network (SON), aimed at making the planning, configuration, management, optimization and healing of mobile radio access networks simpler and faster. Since Release 8, 3GPP has begun the research and standardization of SON, the motivation being to deal with more parameters in network configuration, more complex network structures and the coexistence of the 2G/3G/LTE network. Newly added base stations should be self-configured to realize the plug and play. Moreover, based on network performance and radio conditions, the base station will self-optimize its parameters and behaviour. When outage occurs, the self-healing will be triggered to temporarily compensate the performance loss before a permanent solution is found.

Standards which allow low OpEX management of ECN and software will also be as well.

7.2 Trust/ decentralized trust

The ISO/IEC 15408 Standard defines trust as "a calculation configuration in which components, operations or processes involved in the calculation are predictable under any condition and are resistant to viruses and physical disturbances".

Trust in this sense means that the services provided by the computer system can be proved to be trustworthy. In other words, the services provided are trustworthy from the user's point of view and this trustworthiness is provable.

Trust computing as defined by the Trust Computing Group (TCG) has the following meaning:

- User authentication: the trust of the user
- Platform hardware and software configuration correctness: the user's trust in the platform environment
- The integrity and legitimacy of the application: the trust in the application running
- Verifiability between platforms: the mutual trust between the platforms in the network

The decentralization trend is driven by the distributed system, for instance an edge computing system, in which no central hub acts, so a new approach to security and trust are needed based on the distributed architecture.

There exists a general view that the blockchain's distributed architecture offers a valid framework for tackling distributed system security and trust challenges. The blockchain is a distributed database that maintains a continuously growing list of records, called blocks, secured from tampering and revision. Each block contains a timestamp and a link to a previous block. By design, blockchains are inherently resistant to modification of the data – once recorded, the data in a block cannot be altered retroactively.

ISO Technical Committee 307 is now dedicated to standardization of blockchains and distributed ledger technologies to support secure and trust interoperability and data interchange among users, applications and systems.

7.3 Credible information

Credible information is crucial for an edge computing system. Credibility of information depends on trust in the system which generates the information. Trust is defined to be "confidence that an operation, data transaction source, network or software process can be relied upon to behave as expected" in IEC 62443-3-3 [63]. IEC 62443-3-3 describes system security requirements for industrial automation and control systems (IACS), and it currently does not list trust as a requirement explicitly. Even though security implies a guarantee of trust, it can be useful to review whether some additional system requirements, e.g. requirements on system integration and operation, are necessary to realize trust in IACS. It can also be beneficial to investigate what additional requirements are necessary when dealing with trust in horizontal edge computing systems.

7.4 E/W communication Standards between multiple ECNs

There are several layers of E/W communication in question:

1) Physical layer: a number of Standards exist for mesh networking via physical layer relay and any of these can/could be used. It might be worth considering how that mesh might be implemented efficiently in wired networks, and also whether the physical radio Standards might be merged with the narrowband IoT protocols for long-range operation.

2) Link layer protocols: here again numerous protocols exist (IEEE 802.1aq in wired, and IEEE 802.15.4-ZigBee [64] or Z-Wave [65] and WIA-PA in wireless). Again a merge with narrowband IoT protocols should be considered to allow mesh operation in narrowband IoT for long range operation.

3) In the autonomous control domain, time-sensitive data must be transmitted within strict bounds of latency and reliability. In the case that E/W-bound communication is required between ECNs in industrial automation, automotive or robotic environments, TSN may be needed to prioritize time-sensitive traffic in crowded networks. TSN is currently under development within IEEE 802.1 and the Deterministic Networking working group of IETF.

4) Data layer: a flexible data ontology, allowing common definition of data types and meanings across the network. This is an area where both Standards bodies and open source may play a role, such as oneM2M and OPC-UA.

The majority of open source work might be concentrated in the area of high level data processing in the mesh by elaborating on existing, proven and recommended Standards (such as MQTT) within an open reference architecture. For example, an overarching reference architecture could employ a lightweight MQTT implementation to accept not only north/south (N/S) but also E/W transactions between modules. This could be implemented as a single queue (all transactions E/W, N/S) or as two queues, one operating for E/W and another for N/S. It can be noted that in the items above a mesh can be implemented at each layer, but it is only with the inclusion of item 3) that an application level E/W communication can be achieved.

Finally a successful implementation of E/W communication depends on implementation of decentralized trust.

7.5 Containerization Standard for embedded systems

Linux containers, Docker for example, offer for the first time a practical path to using virtualization on embedded devices, as the latter do not require a very complex hypervisor architecture to operate. Containerization of IoT applications, particularly at the ECN level, would be greatly facilitated by the creation of a common Standard for virtualisation support on IoT nodes. This would be an expansion of the ground covered by OCI [60], which has initiated a general effort.

There are a number of challenges facing an implementation:

- The extreme heterogeneity of device type
- Severely restricted resource envelopes in terms of storage, CPU, and networking

- Devices that are difficult to reach or re-provision upon failure, where power is unstable and may be turned off at any time, or which have custom hardware attached, requiring deep version interoperability. i.e. when the device returns online after weeks or months, an upgrade to the container can be made spanning several versions

Given the level of activity in open source in this area, for example ResinOS [66], see Figure 7-2, it seems that it would be useful to develop such an implementation, or group of implementations, into a Standard defining:

- core kernel services in the host OS;
- core device implementations, and what basic devices would be supported;
- which initial builds for which mixes of processors/peripherals;
- the close linkage between kernel, devices and the containerization framework;
- choice of containerization framework to be the support, for example ResinOS chose Docker.

7.6 Open standard for implementation of algorithm for machine learning

As noted previously in section 3.3.4.6, the complexity of CNNs, HMMs, natural language processing and other disciplines used in the creation of ML algorithms and DNNs requires storage and computing resources usually only accessible on a data centre scale consume considerable power and are relatively expensive. Clearly the backend processing in embedded devices is currently an open source initiative, and since it has started in this manner, it is likely to remain so, with Caffe and a few others becoming *de facto* Standards.

To implement ML upon lower powered, cheaper, embedded devices, it would seem to be a reasonable approach to implement a specific hardware-based method of accepting the introduced ML models and then acting upon them, i.e. comparing the models with incoming live data.

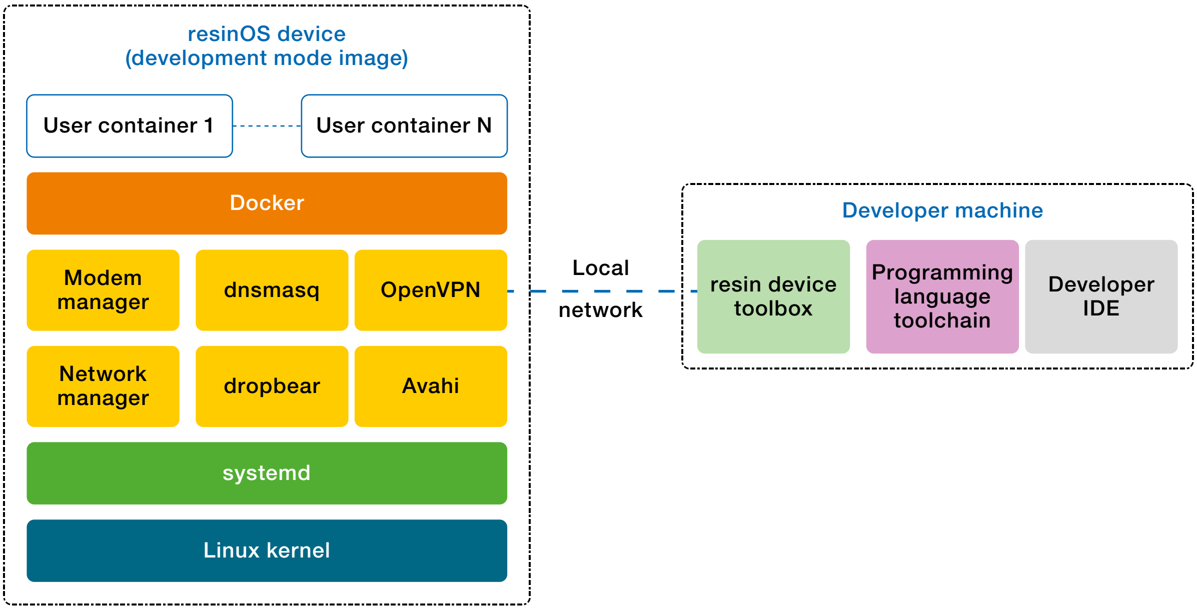


Figure 7-2 | ResinOS embedded [66]

Already some efforts have been made in this area, for example the recent Intel Quark implementation of comparison functions [59]. These efforts are proprietary and no Standards have been defined to cover the loading and comparison of features.

If Standards were defined in the loading of models and comparison of data, it would provide the greatest degree of interoperability between different offerings from different processor manufacturers.

7.7 Comprehensive standard tackling carrier mode selection in case of loss of connectivity

Connectivity is offered by many providers of mobile but also Wi-Fi networks. Some even have global coverage or globally scattered coverage, for example iPass, a network of Wi-Fi access points across the globe, or Eduroam that has Wi-Fi access points across universities.

Currently human users can select the network and input the credentials. Some locations, e.g. hotels, offer a QR code for the credentials.

In the case that the connection is lost, the human has to intervene and connect to a new network. For IoT or safety and security use cases, this interaction is not possible or even productive. There are Standards for sending recommendations regarding which networks to use, with associated policies, for example Open Mobile Alliance (OMA) Device Management, based on HTTP. The Standard was adopted by 3GPP on the interface between the UE and the ANDSF network component. It supports recommended network policies depending on the time of the day, the location and the prioritized networks to be connected. The UE can thus connect independently to a new network when connectivity is lost. Unfortunately the policy does not include the very important aspect of price. Being a Standard oriented to the telecommunications industry, it did not reach out to the outside community.

For tackling IoT use cases, OMA has defined a new protocol, OMA Lightweight M2M that uses a more energy-efficient transport based on UDP. Its connectivity management policies are developing and it might have a broader impact on providing connectivity without human interaction.

Even so, there is a tremendous need across the vertical sectors to have a comprehensive standard tackling carrier mode selection, according to the connectivity modules built on the end device, be it Wi-Fi, 3G, LTE, soon 5G or any other type of access network.

7.8 Role of open source

Cloud computing has immensely benefitted from open sources such as Linux, Docker containers, Kafka messaging, Spark streaming and multi-tier storage. The result has been a highly scalable and standardized infrastructure that meets computational and lifecycle management demands and provides a common environment for developers, driving down the cost of software solutions.

The need for standardization and open source for the edge is even greater. The edge is where vendor-specific solutions need to interoperate. Without this interoperation, IoT cannot fulfil its promises.

As discussed in section 4, microservices or pods need to be deployed on the edge (devices, IoT gateways, micro data centre, etc.) as well as in the cloud, so that applications can be configured in an optimal way, e.g. to address huge data volumes, real-time requirements and variances in connectivity.

As history has shown, open source projects fulfil these needs better than standardization of interfaces and architectures.

Companies providing solutions in edge computing will have plenty of room for differentiation and revenue generation by providing differentiating functionalities, domain specific solutions, better services, higher QoS, etc.

At the time of the writing of this White Paper, the Linux Foundation project Edge X Foundry appears to be a candidate to address a common edge computing platform.

Section 8

Edge intelligence use case testbed

Testbeds are used as a platform for conducting rigorous, transparent, and replicable testing of scientific theories, computational tools, and new technologies. This section describes the testbed initiative that will embody the new concepts of EI to demonstrate the feasibility of the technology. This testbed includes a horizontal testbed platform that implements some of the needed capabilities described in section 6. The same horizontal testbed platform can be used to build vertical testbeds such as one which is a fusion between use cases in the security arena (cameras and access control), ERP and manufacturing systems. The intent is to demonstrate a unified, efficient, portable information flow, see Figure 8-1.

To provide for increased visibility and sustainability, the testbed is intended to be a living platform, with extensions being added as new use cases or new solutions require them.

8.1 Potential use cases which can be implemented on the testbed

- CNC/stock control informs ERP of replenishment requirements.
- An order is placed.
- The ERP system informs access control/ camera system of the expected arrival of addition material, with estimated arrival time/ date, and transport details.
- The ECN system:
  - recognizes a delivery is incoming and issues a warning (at distance);
  - camera (or alternate camera/sensing system) identifies shipment type;
  - informs ERP system of impending arrival.

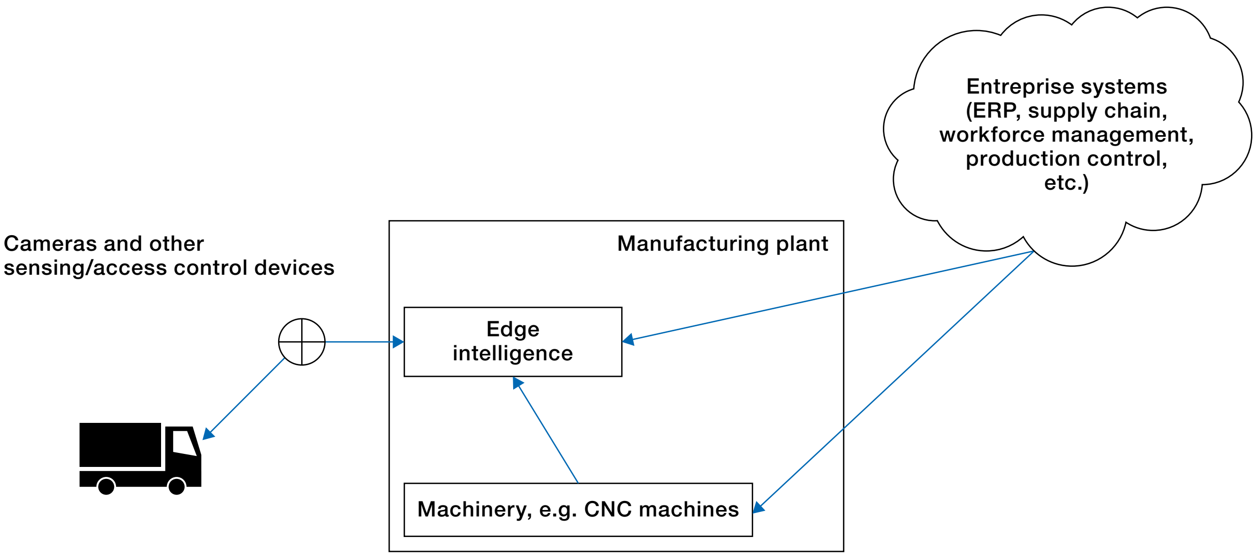

Entreprise systems (ERP, supply chain, workforce management, production control, etc.)
Cameras and other sensing/access control devices
Manufacturing plant
Edge intelligence
Machinery, e.g. CNC machines

Figure 8-1 | Testbed diagram

- Based on production line requirements, the ERP system issues instructions to store or queue the incoming delivery to the destination machine.
- A wayfinding system guides the transport of the delivery either to production line or stores.

8.2 Technology building blocks

To fully accommodate the assertions made in this White Paper, the testbed should employ at least:

- containerized ECNs (technology may not permit containerized sensors);
- ML-based delivery recognition employing advanced or multiple sensors;
- microservice-based interconnections between systems;
- the 5G system as a carrier technology for software-defined networking and identity management.

8.3 Future developments

One of the potential enhancements of the testbed involves adding decentralized trust to facilitate the path towards pre-commercial solutions, as well as having multiple dedicated networks for supporting either different use cases or networks from the same use case that need isolation for security reasons.

Section 9

Conclusions and recommendations

9.1 Conclusions

The following conclusions can be drawn from the review and analysis undertaken in this White Paper concerning both the technological potential of edge intelligence in implementing the IoT and the current gaps and requirements, including in the area of standardization, that need to be addressed to realize that potential.

- Edge intelligence is edge computing with **machine learning** capabilities. This means that data can be analyzed and decisions can be made by algorithms at the edge, i.e. very close to where the data is collected and where the machine and other equipment is controlled. This makes it possible to react autonomously, without a connection to the cloud, and with very short response times.
- **Containerization** will be important for edge intelligence. Containerization allows to encapsulate functionality, for example a machine learning algorithm, in a software package, the so-called container, which can be deployed anywhere, e.g. in a public cloud, a private cloud, on premise, a micro data centre on a shop floor, in a vehicle, within a 5G network or on an IoT gateway. This increases the efficiency of software development and allows to optimize the deployment according to customer specific requirements without additional programming effort. There currently exist no Standards directly covering this technology, although there are many open source initiatives, such as Docker and OCI.
- **Common data models** for edge computing node communication are essential to the success of edge intelligence. A common data model enables the interoperability between devices, communication protocols and software solutions from different vendors.
- **Micro data centres** will become more important in this process, for a number of reasons, including providing low latency and processing large volumes of data, thus avoiding transportation of such data to the cloud, which can be impracticable or costly.
- **5G networks** will provide data centres at the edge as well as the possibility to implement industry-specific networks enabled by virtualization and software-defined networking principles. This allows customers to reduce their costs and increase their efficiency in a manner similar to the benefits provided by cloud computing, as customers do not have to deal with provisioning and maintaining data centres at the edge.
- **The best user interface is no user interface**. Traditionally, user interfaces to enterprise systems enable human users to input and analyze data and to execute decisions. IoT makes manual data input largely obsolete, as data is collected automatically. Machine learning and artificial intelligence take over the data analysis and decision-making. Human interaction and interference are largely reduced.

9.2 Recommendations

The IEC should call for, take initiatives concerning, invite and strongly contribute to a more global and collaborative approach to the implementation of edge intelligence, involving not only international standardization organizations but also consortia in the edge intelligence landscape. The recommendations listed below are directed at industry, future standardization actors across many sectors, and the IEC itself.

9.2.1 Industry-targeted recommendations

The following recommendations are addressed to the various industries that will be impacted by the development and implementation of edge intelligence applications:

- Prepare for disruption of business and commercial models. During the last decade, we have experienced a change from the traditional software license model to the services model: software as a service, platform as a service, infrastructure as a service. These services are typically located in the cloud, i.e. in centralized data centres. With the advent of edge computing, we will see an extension of these service models to the edge and combinations of traditional license and service models.
- Utilize 5G Standards to facilitate edge computing and edge intelligence solutions. 5G networks have the potential to provide benefits similar to those of cloud computing, but focused on the edge instead of in centralized data centres.
- Include micro data centres in edge intelligence solution architecture, ideally employing containerization. Micro data centres can provide extremely short response times and reductions in communication-related costs. Containerization provides the flexibility to optimize deployment of functionality according to specific customer requirements without additional programming effort.
- Agree on a common approach to orchestration and lifecycle management and to machine learning (tools, model implementation) to avoid market fragmentation. Their commoditization will drive down cost.

9.2.2 General recommendations

There currently exist many standardization activities and consortium activities related to edge computing. This situation provides challenges to optimizing edge computing standardization and opportunities for creating a more positive standardization ecosystem that supports the needs of governments, the private sector and users. This ecosystem should be one of collaboration across the spectrum of SDOs and consortia as outlined below.

- Horizontal standardization: International Standards should be the preferred approach for standards activities that cross domains, geopolitical boundaries, functionalities and requirements elaborated at the international level.
- Vertical and specialty Standards: Standards that are domain-specific or geopolitical should come from relevant organizations. Wherever possible, they should draw on higher-level horizontal Standards.
- Requirements for edge intelligence Standards: leading consortia should define requirements and feed those requirements to existing Standards bodies.

All SDOs, consortia, geopolitical entities and other entities involved in edge intelligence definition, development, deployment and operation should publically adopt the guiding principles, as described above, and should look for opportunities to foster increased levels of cooperation and collaboration in their implementation.

9.2.3 Recommendations addressed to the IEC and its committees

The IEC Market Strategy Board (MSB) is responsible for identifying technology trends and market needs. This can be achieved in different ways:

- White Papers, articulating the IEC viewpoint on new technologies and market changes
- Testbed projects, providing industry feedback on IEC International Standards from a technology and market perspective
- Interoperability projects between Standards and open source-based implementations, bridging the open source and Standards markets/communities

The IEC is in a unique position to drive edge intelligence forward. Accordingly, the IEC should take the following actions:

- Following extension of software implementations within electrotechnical systems, take a higher profile in promoting the software component of this domain.
- There is a strong need for edge intelligence Standards, as described in section 7. These should be validated via a testbed, which may identify needs for further improvement.
- Review the findings and recommendations contained in sections 5, 6 and 7 and identify specific activities to be undertaken by the IEC Standardization Management Board (SMB).
- Drive edge intelligence in ISO/IEC JTC 1/SC 41 and IEC SEG 8, as described in sections 7.1 and 7.4.
- Foster an in-house environment for reception of industry feedback on IEC International Standards as well responding to open source developments on key technologies.

  Industry feedback is critical to indicate technology/Standard gaps needed for the “desired” industry transformation and to accommodate the evolving business models. This focussed orientation will also indicate the readiness of IEC to support market needs through its International Standards. In addition, the operational and information technologies are converging, and IEC readiness to embrace open source will become more and more critical. IEC needs to respond to the development of open source projects on key technologies within the market.
- Collaborate with the Industrial Internet Consortium (IIC) to complement Standards with open source implementations and testbeds. However the IEC MSB should manage this interaction with the IEC/IIC testbed initiative, e.g. by means of a dedicated project management team.
- IEC/IIC testbed projects should be forward-looking with regard to new technologies or business models. This activity should not disrupt ongoing specification activities within the IEC SMB, the IEC technical committees and subcommittees, and ISO/IEC JTC 1.
- The objectives of the IEC/IIC testbeds should be to identify technology gaps within the industry transformation, identify specific market needs for Standards and address interoperability between Standards and open source-based implementations.
- The IEC/IIC testbed projects should communicate their findings in the form of project reports to the IEC to enable IEC SMB steering of the standardization activities accordingly.

Annex A

Use cases and requirements for edge intelligence

A.1 Factory productivity improvement

A.1.1 Scope

- Domain: smart manufacturing
- Architectural levels: device, edge, cloud

A.1.2 Objectives

- Improved productivity and profitability of manufacturing equipment
- Reduced downtime of manufacturing equipment

A.1.3 Narrative of use case

A.1.3.1 Summary

A combination of edge computing and cloud computing realizes improvement of productivity and reduction of downtime for a factory.

A.1.3.2 Nature of the use case

Edge computing converts raw data collected from real world to data which is useful for analysis using context information.

A.1.3.3 Complete description

Big data analysis realized by the combination of edge computing and cloud computing with e-F@ctory solution [57] can greatly benefit the operation of a factory.

An edge computer in a semiconductor factory processes sensor data collected from the shop floor and converts it to data which can be analyzed by a server in a cloud. With the help of the analysis result, factory operators can minimize the number of products determined to be defective by mistake and improve the productivity. The analysis result also enables predictive maintenance.

A.1.3.4 Diagram of the use case

See Figure A-1

A.1.4 Use case conditions

A.1.4.1 Assumptions

Factory operators cooperate with system integrator to identify what data to collect and how to analyze it so that the objectives can be achieved.

Necessary data can be usually gathered by installing additional sensors, if necessary, without modifying the factory equipment.

A.1.4.2 Prerequisites

None

A.1.5 Further information for the use case

A.1.5.1 State of the art

Status quo:

- Improvement of productivity is effected based on human experience. Improvement is usually closed in one factory.
- It is difficult for factory operators to understand what to do with data obtained from the shop floor.

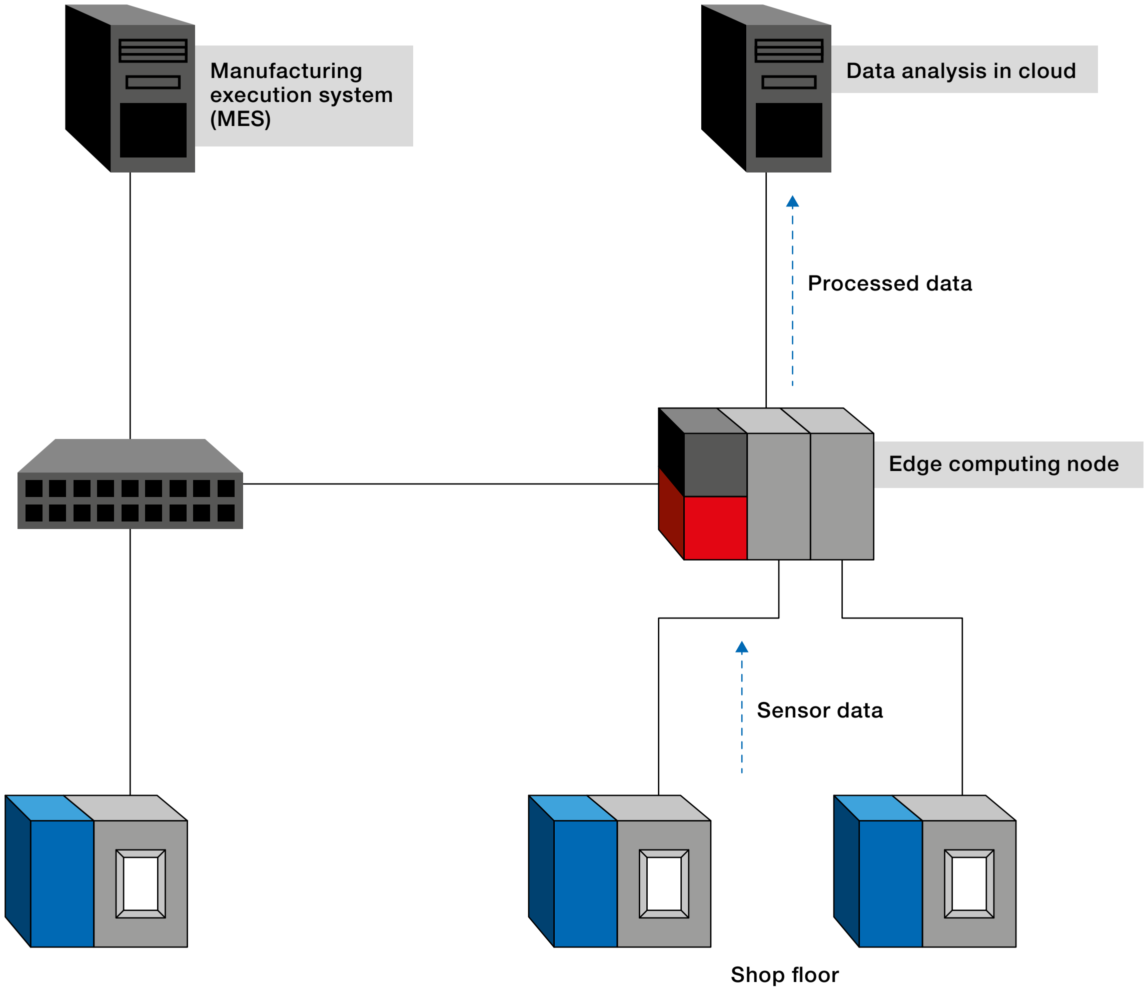


Figure A-1 | Factory of the future use case

State of the art:

- Factory automation systems can help factory operators improve the productivity by visualization and analysis of factory operation data.
- Visualization and analysis can be done across multiple factories.

To switch status quo to state of the art, technologies/Standards to promote the adoption of edge computing systems are needed.

A.1.5.2 Technology gaps

- Credibility of information generated by edge computing
- Assisted/automatic optimization of system operation
- Edge computing environment which allows applications to be deployed

A.1.5.3 Needed capabilities

Management of secure identity:

- as an enabling technology for credibility of information generated by edge computing.

- e.g. PUF.
- ensuring that information is not hampered by unauthorized parties during the communication using secure identity.

Machine learning technology:

- as an enabling technology for automatic optimization of system operation.

A.1.5.4 Necessary future Standards

Definition of basic functionalities to be provided by an edge computing platform, e.g. data acquisition from real world, data management and communication between applications.

Standards to improve the interoperability of edge computing applications:

- Specification of external interface of the functionalities provided by an edge computing platform
- Protocol for inter-application communication
- Semantic interoperability of data

A.2 Connected city lighting

A.2.1 Scope

- Domain: urban IoT, smart city
- Architectural levels: end devices/sensors, edge controller/gateway, cloud
- Sub-use cases:
  - Smart lighting
  - Environment monitoring
  - Utility management
  - Video surveillance
  - Traffic management
  - Vehicle charging
- Super-use cases:
  - Smart city/campus

A.2.2 Objectives

- Reducing energy consumption in city lighting
- Avoiding repetitive construction, saving urban space
- Improving efficiency of municipal management, reducing OpEx

A.2.3 Narrative of use case

A.2.3.1 Summary

Street lighting is considered to be one of the most important landing points of urban IoT. Energy consumption can be reduced by introducing flexible on-demand lighting. Moreover, the lighting poles can be used as IoT access nodes to integrate a variety of functions, which provides smart city development with valuable big data and integrated interaction to improve citizens' lives.

A.2.3.2 Nature of the use case

Edge computing deals with the data based on which timely responses are expected.

Edge computing conducts pre-treatment of the raw data, e.g. aggregation, filtering, to offload the burden at the cloud.

A.2.3.3 Complete description

Global electricity consumption totals 20 450 billion KWh, of which 19% is attributable to lighting power consumption, see Figure A-2.

Although upgrading the traditional high pressure sodium lamp (HPSL) to the latest light-emitting-diode (LED) lamp is quite power effective, the consumption per day for each lamp could be reduced from 4,8 kWh to 1,8 kWh. Some challenges to this improvement still need to be overcome:

- Huge energy consumption
- Lamp cannot adjust the brightness or switch as the weather condition changes, so a lot of energy is wasted
- High OpEx
- 352 million street lamps are expected by 2025
- Closed architecture
- Hard to evolve to smart city pattern

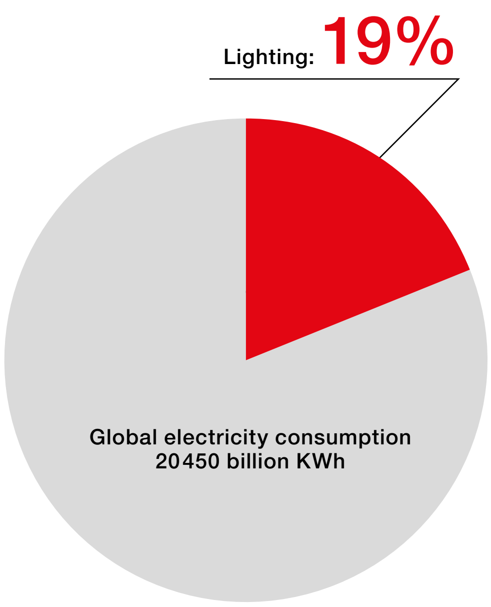
To address these issues, connected city lighting is proposed, which has the following advantages:

- Smart lighting policies

  The basic lighting duration and brightness can be adjusted according to the exact time, date and local latitude. For example, reduce street lamp brightness or only switch on every other lamp at midnight, and shorten the light duration in summer at high latitude areas. In addition to this basic lighting plan, on-demand plans can be added. For example, turn on lamps when the visibility is poor on rainy days, or increase the brightness when vehicles or pedestrians approach, which is safer for citizens.

- Local survival mechanism

  Traditional streetlights are controlled in a centralized fashion. A failure can turn lights on in bright daylight, which wastes large amounts of energy. Connected city lighting could avoid such situations. In the case that the connection to the remote control in the cloud is unreachable, the gateway will use local cached policy to control the streetlights, thereby increasing flexibility and reliability.



Source: Index mundi

Figure A-2 | Global electricity consumption [58]

- Visible management and online inspection

  A geographic information system (GIS)-based visual management system can be adopted to monitor the lamps' status in real-time. When a lamp malfunctions, the system can be alerted automatically and the maintenance staff can be informed. This reduces the need for on-site inspection, helps lower the labour costs and improves management efficiency.

- Service life statistics

  Predictive maintenance can be enabled by comparing the status of the lamp and the statistical data. When a lamp's service life is about to finish, the system can remind the staff to replace it.

Connected city lighting lays a solid foundation for urban IoT. As a further step, light poles would be modular so that various function modules can be easily added on as needed. Thus the pole is not only for lighting usage but constitutes an integrated system of sensing and service provision. In this way, unnecessary construction of poles or equipment cabinets is avoided, which helps save roadside space.

Possible modules could include, but are not limited to, the following:

- Environment monitoring module: sensors such as temperature, humidity and air quality
- Wi-Fi hotspot
- Video surveillance module
- Advertising module: LED screen, speaker
- Charging facilities for electric vehicles
- Municipal management module: gateway and edge controller for distributed sensors deployed in e.g. rubbish bins, parking spaces, underground pipes, etc.

  For example, EI can improve the efficiency of rubbish collection. Currently, sanitation workers have to pass by each bin to check the bins periodically. Without an *a priori* knowledge, sometimes they pass by only to find the bin is empty. Detailed information such as trash quantity and type can be acquired by sensors installed inside the bins, and the right type of vehicle can be sent accordingly for collection. An optimized schedule and path can also be planned. This on-demand collection can help reduce the OpEx of sanitation companies.
- V2I device

  Integrated with the V2I communication device, the pole can become a roadside unit (RSU) in the vehicular network environment. It can be used for exchange of critical safety and operational data between vehicles and roadway infrastructures to guarantee vehicle safety. V2I devices can also work as beacons to increase the positioning accuracy which is required by autonomous driving. Information on vehicles such as position or velocity can be gathered via V2I, so that the edge controller is able to find an optimized traffic light signal plan to let more vehicles pass through intersections. Alternatively, a suggested speed can be given to vehicles to avoid encountering a red light.

A.2.3.4 Diagrams of the use case

See Figures A-3 and A-4

A.2.4 Use case conditions

A.2.4.1 Assumptions

Lighting becomes just one of the functions of the utility pole, and the applications built on the edge OS and cloud OS belong to different organizations, e.g. utility companies, public safety agencies, advertising companies, property companies, etc.

A.2.4.2 Prerequisites

Sensors, communication devices and other modules can be added on the existing poles without great modification of the poles' structure.

New modular poles can be designed and built for new infrastructure construction to further facilitate the installation.

A.2.5 Further information for the use case

A.2.5.1 State of the art

In this use case, the pole integrated with functional modules acts as a sensing station and an access to various services in the urban scenario. In a city scale, tremendous data can be taken advantage of for deep analysis and visualization in the cloud. At a smaller scale, such as a street, a domestic area or a parking area, EI can bring convenient and timely services.

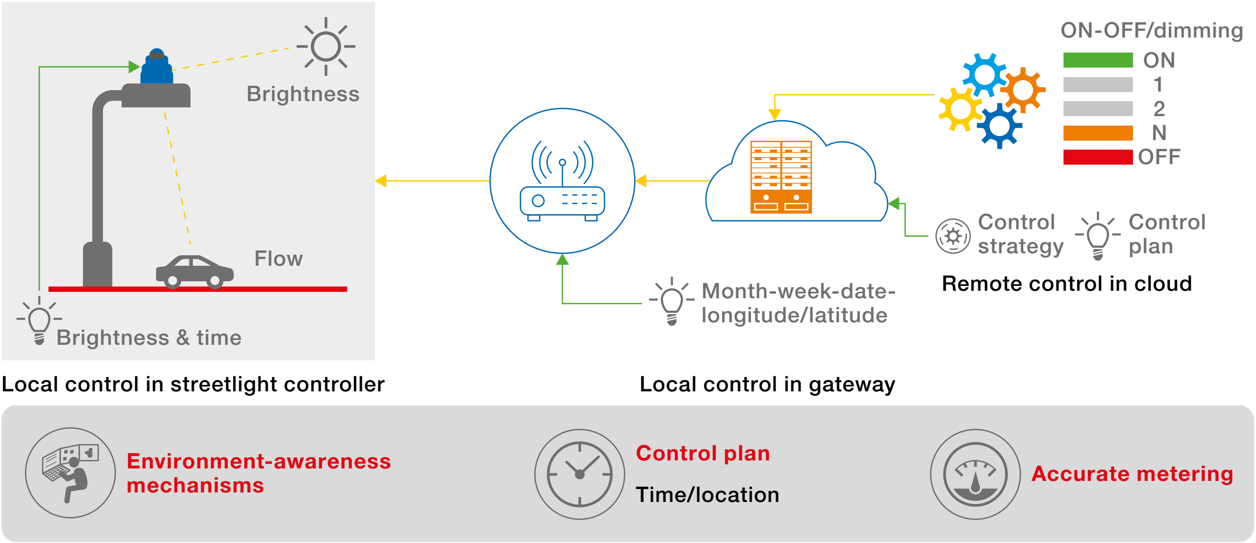


Figure A-3 | Connected city lighting use case [source: Huawei Technologies]

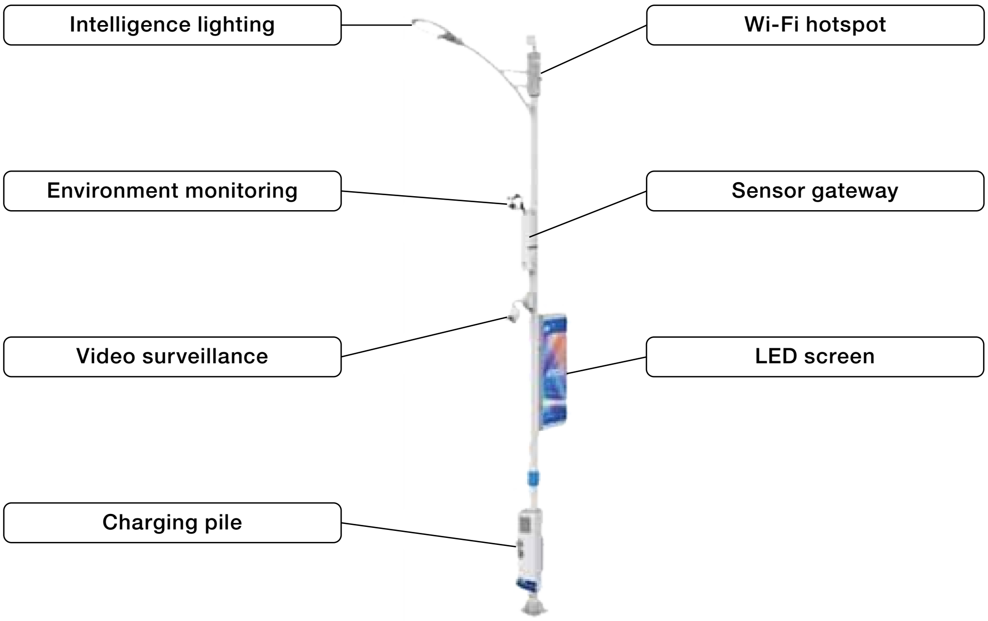


Figure A-4 | Intelligent utility pole [source: Huawei-Sansi]

A.2.5.2 Technology gaps

- Pole networking, function add/remove: the current IoT applications are built in silo mode, and applications belonging to different organizations have their own end-to-end solutions, which may lead to exponential increase of capital expenditure (CapEx) and OpEx. As the devices may be deployed progressively over a long period, backward and forward compatibility should also be considered.

- Evolutionary lighting policies and energy management policies: the policies can evolve over time to provide better and better services.
- Cloud offloading and privacy: some applications such as video surveillance will produce large amounts of data. Storing and processing the data at the cloud brings pressure to the network; moreover privacy may not be guaranteed if the data is not stored in the local area.
- Interaction between ECNs: some applications could not be realized by only one ECN, therefore interaction between nodes should be introduced to conduct user information synchronization, control and orchestration.

A.2.5.3 Needed capabilities

- Self-organization, self-discovery, plug and play of functional modules: when adding/removing a module, the related application is added/removed from the platform. Thus the modules can be managed in an agile way and repetitive development is avoided.
- AI for ECNs to support learning abilities.
- The ECNs should have enough processing and storage capabilities to support the local data processing requirement.
- E/W-bound communication between ECNs.

A.2.5.4 Necessary future Standards

The end devices can be produced by different manufacturers, but they should be transparent to edge and cloud applications. Adaptations between the IP and various hardware technologies must be introduced. The 6LoWPAN Standard has been proposed by IETF to build an adaptation layer for IEEE 802.15.4, and it can also be used for sub-GHz ISM and PLC. 6LoWPAN primarily considers the fragmentation and header compression. Currently, other IETF working groups are studying other IoT-related problems, e.g. ROLL studies the routing protocol for low-power and lossy networks, and LPWAN studies IPv6 over low power wide area networks. As for the V2I communication which might be implemented in this use case, the IPWAVE working group is studying how to build direct and secure communications in vehicular environments.

Standards for self-organization, self-configuration and self-discovery: The end devices can be produced by different manufacturers using different PHY and MAC technologies, but they should be transparent to the operators and the upper layer applications. How to realize the plug and play of various devices managed under one and the same platform must be researched.

E/W communication Standards should be considered not only in the physical and link layers but also in the data layer. To realize cross-vendor data interoperation and analysis, unified semantic meanings are required.

Containerization Standards are necessary to realize virtualization on embedded devices and will help break the silos of different IoT services and avoid repetitive development of applications.

Open Standards are needed for ML.

Trust and security: related Standards should be developed to prevent tampering or revision of data or even illegal control of ECNs.

A.3 Smart elevator

A.3.1 Scope

- Domain: urban IoT, smart building
- Architectural levels: devices/sensors, edge controller/gateway, cloud
- Sub-use cases:
  - Advertising inside the elevator
  - Video surveillance
- Super-use cases:
  - Smart city
  - Smart building/campus: the smart elevator can be one of the applications installed on a platform of building/campus management used by a property company

A.3.2 Objectives

Urbanization makes elevators indispensable in cities, see Figure A-5. It is self-evident that elevator maintenance and after-sales service are huge business opportunities. More elevator vendors are integrating industry chains and increasing revenue by providing O&M services. However, traditional maintenance costs still remain high, and the first maintenance success rate is below 20%. Therefore, to improve O&M efficiency and ensure lower O&M costs, digital transformation must be introduced into traditional elevator maintenance.

A.3.3 Narrative of the use case

A.3.3.1 Summary

Edge computing can help elevator vendors upgrade from the traditional preventive maintenance to next-generation real-time predictive maintenance, extending value from products to services.

A.3.3.2 Nature of the use case

The duty division and cooperation between the edge and the cloud: the edge is in charge of scenarios where a timely response is needed, e.g. fault detection, and the processed raw data is sent to the cloud for further analysis and AI training.

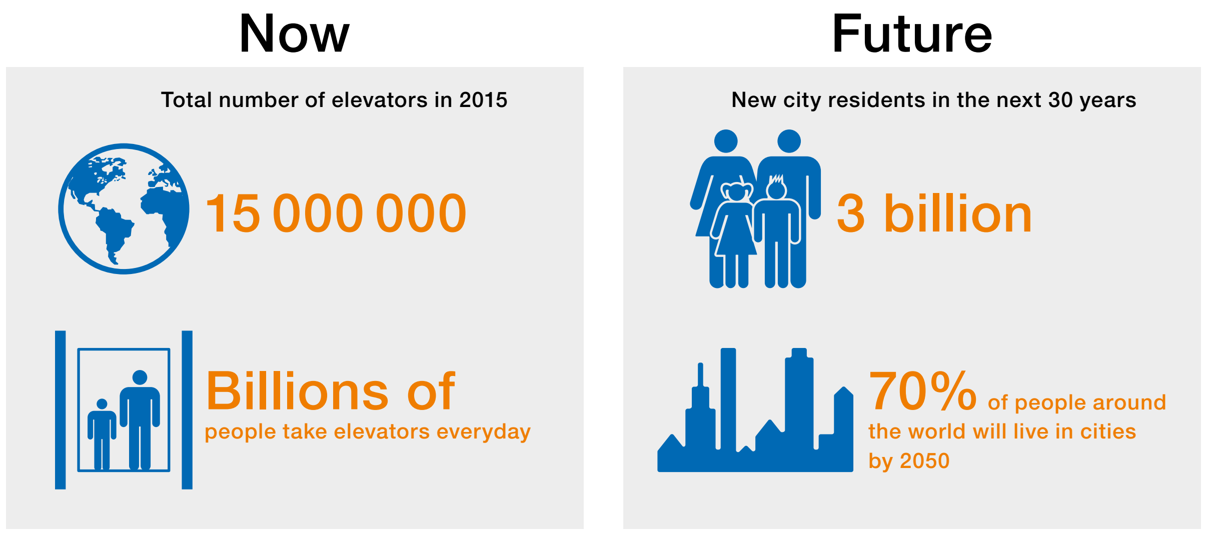

Now
Total number of elevators in 2015
15 000 000
Billions of
people take elevators everyday
Future
New city residents in the next 30 years
3 billion
70% of people around the world will live in cities by 2050

Figure A-5 | Predictions on elevator business use case [source: Huawei Technologies]

A.3.3.3 Complete description

- Reducing costs

  A large number of sensors can monitor the elevator's status in real-time. A local edge computing converged gateway can provide data analysis capability that detects potential device faults early. This model provides local resilience. If the gateway fails to connect to the cloud, data can be stored locally. After the connection is restored, the stored local data can be synchronized to the cloud automatically to ensure that the cloud can generate a complete view of each elevator. Predictive maintenance can reduce the labour workload, strengthen device reliability to prolong service life, improve device utilization, and thus cut maintenance costs. All these capabilities lift the overall competitiveness of enterprises.

- Security assurance

  Predictive maintenance provides multiple-level protection that covers terminal devices, gateway chips and OSs, networks and data.

- Product-to-service extension

  Elevator vendors' research and development teams can improve their product quality and after-sales services. With predictive maintenance, building owners and property management agencies can provide emergency rescue services. Further, elevators can serve as media platforms for advertising.

A.3.3.4 Diagram of the use case

See Figure A-6

A.3.4 Use case conditions

A.3.4.1 Assumptions

None

A.3.4.2 Prerequisites

- Digital model: mathematical or mechanism models of elevators must be built.
- Status acquisition: the data acquired by the sensors should be sufficient to represent the real status of the elevator, thus both edge computing and cloud computing can have correct outcomes.
- Digitized rules: the duty division between the edge and the cloud should be defined. The local diagnostic needs simple but critical rules to detect malfunctions. The corresponding responses to the malfunctions should also be given.

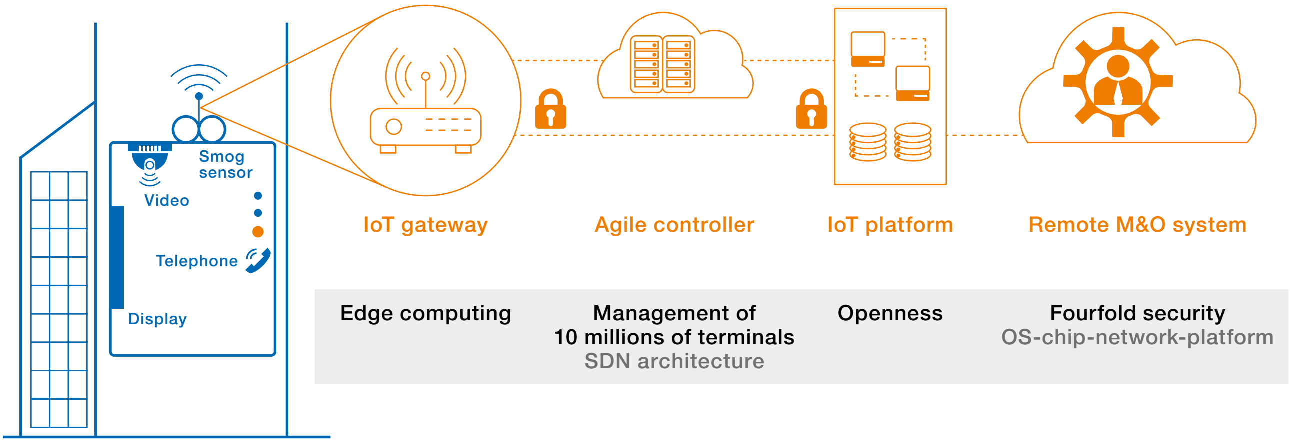


Figure A-6 | Smart elevator use case [source: Huawei Technologies]

A.3.5 Further information for the use case

A.3.5.1 State of the art

The gateway, the agile controller and the platform are software-defined; containers can be provided to vendors to add their own O&M applications. In addition vendors can also provide some containers to building owners and other clients for value-added services such as advertising and notices.

A.3.5.2 Technology gaps

- Diagnosis and predictive maintenance: since the elevator is strongly related to public safety, the fault detection and maintenance is considerably important. Currently, most faults are detected after they occur, and it often takes a considerable amount of time to identify the fault and repair it. Therefore the OpEx in elevator maintenance is very high.
- Task partition between cloud and edge: functions that demand a timely response such as fault detection should be deployed at the edge and operate even without connection to the cloud.
- Interaction between ECNs: required for optimizing the behaviour of elevators, in the scope of management strategy.
- Security in preventive maintenance: the field engineers must be authenticated before gaining access to elevator systems.

A.3.5.3 Needed capabilities

- AI: the training can be conducted at the cloud, and the outcoming strategies can be embedded at the edge, so that the edge can respond in a timely fashion.
- The ECN should have enough processing and storage capabilities to support the local data processing requirement.
- E/W-bound communication between ECNs is necessary.
- Decentralized authentication mechanisms are required in order to prevent tampering or revision of the edge and could contain information.

A.3.5.4 Necessary future Standards

The basic and necessary features used in diagnostic and predictive maintenance need to be standardized for different categories of elevators in industry Standards organizations.

Standards for self-organization, self-configuration and self-discovery: the elevators and devices inside stem from different manufacturers; however, they should be transparent to the operators and the upper layer applications. Automation in organization, configuration and discovery will help realize the plug and play of the elevator systems, which could bring lower OpEx.

E/W communication Standards should be considered not only in the physical and link layers but also in the data layer. To realize cross-vendor data interoperation and analysis, unified semantic meanings are required.

Containerization Standards are necessary to realize virtualization on embedded devices, so that applications do not have to be developed separately for different vendors.

Open Standards for ML are needed.

Trust and security: related Standards should be developed to protect private data and prevent illegal control of elevators in order to guarantee public security.

A.4 Indoor location tracking

A.4.1 Scope

Potential areas of application include medical, office environments where sensitive information is held, military, law enforcement, healthcare and life safety.

A.4.2 Objectives

To provide accurate, cost-effective information on the location of persons and objects.

A.4.3 Narrative of the use case

A.4.3.1 Complete description

- Name of the use case: person tracking – indoor applications
- Using radio trilateration, and/or radio surface mapping mechanisms across a variety of radio systems (BLE, etc.) to get detailed location information for a person/object being tracked.
- Requirements for bandwidth: moderate (approx. 2 MB, very low latency <1 ms, low contention).
- System requires backhaul of trilateration data for a number of sensor sources (all normalized to IP/UDP packets) and conversion into a high quality location estimate.
- Since latency is a real problem with a system of this type, having the gateway which processes the sensor samples as close as possible to the source is critical, while still maintaining the gateway in the cloud, or at least outside the customer premises. Within the gateway module is an embedded, tunable, MI module to perform the location estimation, which then forwards real world positions and user status to the administrative/UI module in the cloud. The model may also be tuned and the MI module updated.

A.4.3.2 Diagram of the use case

See Figure A-7

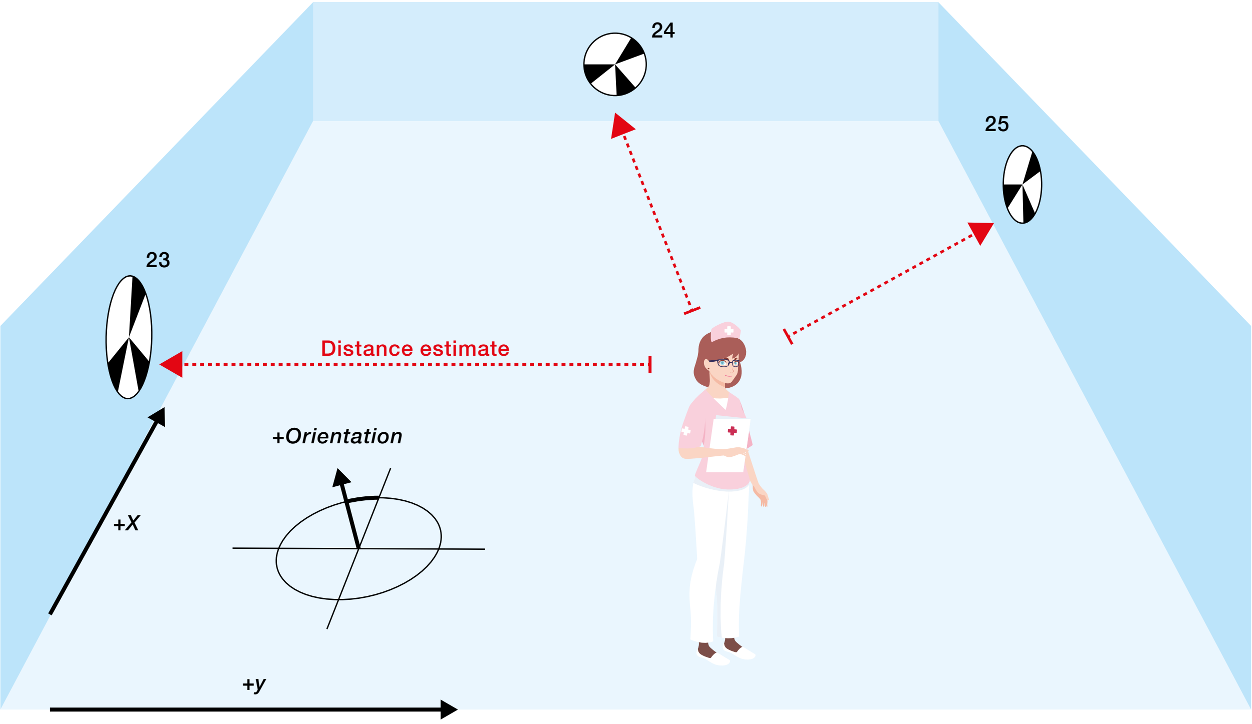


Figure A-7 | Hospital staff/asset tracking use case

A.4.4 Use case conditions

A.4.4.1 State of the art

A number of different solutions are already on the market for this type of tracking, but the focus changes as each technology advances a little. The most advanced systems indoors currently employ ultra-wideband (UWB) or BLE wireless, though very advanced applications have been using geomagnetic sensors and sonar. But these can be considered to be niche applications, as well as the Wi-Fi-based system, which offer a less accurate result.

A.4.4.2 Technology gaps

Accurate tracking is based on two basic tenets: a high local network bandwidth (with reasonably low latency <100 MS) and provision for backhaul to a local processing node with sufficient compute to allow the calculations necessary. When multiple targets are being tracked, then the bandwidth requirement increases linearly. The inability to add this capacity seamlessly within existing and future networks is also a key gap, as currently the only way to achieve this is to add equipment to existing networking infrastructures to operate tracking.

Additional to the above, the local processing node should be capable of a high degree of localized processing, at the level of what would be (in 2017) a server class machine: ideally the processing logic should be containerized, or modularized in such a way as to operate seamlessly in spare space on general IT equipment, or if developments in embedded hardware allow, a gateway scale compute node.

A.4.4.3 Needed capabilities

Bandwidth increases within both public and private radio networks (see 5GPP), as well as capability to hand back and forth seamlessly between these networks, will allow tracking networks of this type to be fully realized.

Increases in embedded compute capacity are necessary to support systems of this type, with localized MI capability (at least at the level of feature comparison) also being useful.

Containerization and self-configuration are required to be implemented at a higher level to allow easy installation. Containerization will allow software to be delivered in discrete, secure, bounded packages, where the API connections can be tightly defined via microservices, without any attendant requirement for library or development language/OS requirements. Containerization will allow multiple IoT applications from different vendors to coexist on processing hardware and to be maintained and upgraded separately with no reference to applications of other vendors or effect on their operation, while allowing information-sharing along pre-defined interfaces, E/W-bound at the device or gateway level.

A.4.4.4 Necessary future Standards

Standardization of "time of flight" systems (such as UWB) and BLE systems would be an advantage, as there are a multiplicity of solutions in the market, all with varying accuracy and bandwidth requirements, with no benchmarks to measure against.

Addition of Standards governing the encoding of feature data in embedded processors [59] would allow greater portability of feature data between different processor types without limiting manufacturers' freedom to innovate.

Self-configuration will be facilitated by a detailed set of recommendations/protocols for discovery, advertisement of device capabilities and registration on the network. Some of these could be based, or related to the emergent 5GPP Standards in this area.

Containerization of IoT applications (particularly at the gateway level) would be greatly facilitated by the creation of a common Standard for virtualization support on IoT nodes. This would be an extension of the ground covered by OCI [60] which has started a general effort.

A Standard governing E/W data sharing at this level between applications using a microservices Standard may be useful.

A.5 Lone worker safety

A.5.1 Scope

Potential areas of application include all lone worker situations as well as those involving workers in sensitive/dangerous outdoor environments.

A.5.2 Objectives

To provide accurate, cost-effective information on the location of persons and objects.

A.5.3 Narrative of the use case

The case involves using an intelligent gateway to receive signals indicating the location of a particular employee. In this case an MI module would also be embedded in the GPS signal transmitter, which would use an MI module to characterize the wearer's gait and orientation. The module would "learn" over a period the "normal" behaviour of an individual, and thus be able to generate an alarm should that behaviour change due to accident, attack etc. Transmission bandwidth would be very low and latency could be in milliseconds, as decision- making would be local. The intelligent gateway in this case would function only as a data consolidator and could also have a UI, etc. installed.

A.5.3.1 Complete description

- Name of the use case: person tracking – outdoor (GPS visible) applications
- Using GPS to get detailed location information for a person/object being tracked, as well as accelerometer data, backhauled through low bandwidth radio.
- Requirements for bandwidth: moderate (approximately 5 kB/day, high latency <1 minute, contention not an issue).
- GPS is processed and requires no post-processing, only a packet frame needs to be forwarded.

- The major consideration for such an application is that instead of deep coverage (i.e. high bandwidth) a very broad coverage is required.

A.5.3.2 Diagram of use case

See Figure A-8

A.5.4 Use case conditions

A.5.4.1 State of the art

Traditionally this backhaul has been effected through cellular networks, but there is an option of employing emergent IoT protocols such as SigFox, or cellular narrowband IoT. High-end systems of this type use a satellite backhaul, but this is not suitable for use with lone workers due to power demands and miniaturization.

A.5.4.2 Technology gaps

Solutions do exist in the marketplace but are artisanal in nature and have not been widely adopted. Equipment is physically quite bulky due to the power/antenna requirements of current systems (either cellular or satellite-based) as well as the power requirements for maintaining GPS operation.

In addition, false alarms are frequent, as the processing power onboard is not sufficient to allow MI to tell the difference between normal movement, or work activity, and an exceptional situation such as a fall, nor is the processing sufficient to allow intelligent power management of the peripheral devices.

However, the primary objection to all of these systems is tied to the backhaul mechanism for

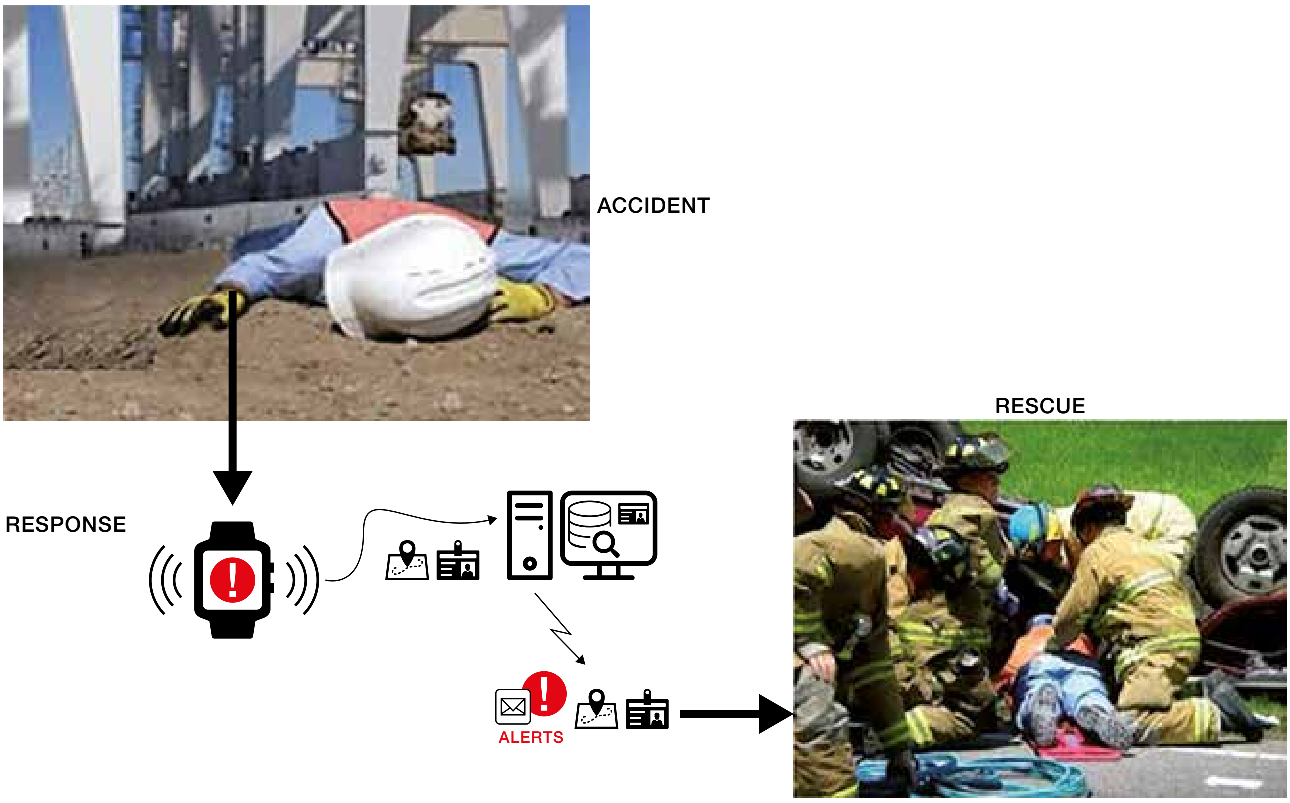


Figure A-8 | Lone worker safety procedures communication of alarm/status data back to the monitoring centre. Firstly, coverage is dictated by either the cellular coverage footprint (patchy in areas of interest for lone workers, particularly in buildings, tunnels, or underground), or access to the sky (satellite communications). Secondly, the cost of these transport mechanisms is not justified by the level of data being transported.

A.5.4.3 Needed capabilities

Narrowband IoT protocols such as Sigfox to reduce signal loss, coupled with deployability on a worldwide basis.

Better battery performance for GPS operation, or lower-power GPS operation. This may also be implemented by intelligent operation of the device to extend battery lifetime.

Increases in embedded compute capacity at lower power are necessary to support systems of this type, with localized MI capability (at least at the level of feature comparison) also being useful.

A.5.4.4 Necessary future Standards

While interest organizations have proposed Standards [61] there has been no industry-wide attempt to standardize the “wide area” IoT access networks, apart from those emerging from cellular narrowband IoT [62].

Addition of Standards governing the encoding of feature data in embedded processors [59] would allow greater portability of feature data between different processor types without limited manufacturing freedom to innovate.

A.6 Access control – tailgating detection

A.6.1 Scope

Potential areas of application include all access control in sensitive areas: banks, offices, hospitals etc.

A.6.2 Objectives

To provide accurate, cost effective door tailgating detection: i.e. that a single authorized entry through an access point has permitted more than one person.

A.6.3 Narrative of the use case

Using an MI module attached to an inexpensive stereo camera to perform head and shoulders detection, based on a starter dataset, to eliminate tailgating at security doors. Coupled with this, an additional reader (BLE/RFID for example) would allow a badge to be read and thus allow frictionless access through the door, i.e. the door opens as the person approaches, but only if a valid signal is read from each user (if there is more than one user) approaching. If more than one person is detected approaching, and fewer badge signals are read, then if those people attempt to go through the door, an alarm will be registered. The inexpensive camera is linked to a processing module, either located in-premise or in the cloud. Latency and bandwidth in this case are both issues.

A.6.3.1 Complete description

- Name of the use case: video detection of access violations – tailgating
- Using a stereo camera with low processing to recognize head/shoulder models and detect whether more than one person is attempting to enter an authorized entry at once (using a card or frictionless).
- Once a count has been effected, a simple notification is made.
- Requirements for bandwidth: high (approximately 20 MB/second, low latency <5 ms, contention an issue) between camera and processing node (if not co-located); or low (1 kB/day if the processing is local).
- It is worth noting here that depending on where the "intelligence" decides that the level of bandwidth the network is required to bear lies, there is a 10^5 scale difference between input and output bandwidth.

A.6.3.2 Diagram of use case

See Figure A-9

A.6.4 Use case conditions

A.6.4.1 State of the art

No broad solutions to this problem are widely deployed.

A.6.4.2 Technology gaps

While this can be achieved currently, both cameras and compute nodes are required to be of a very high standard, and a solution is needed to operate on low power equipment with more basic cameras. Coupled with this issue, it is challenging to install solutions due to the need for calibration to an environment. Additionally the camera is required to be on a high-bandwidth, low-latency network to allow very fast communication with the compute node.

A.6.4.3 Needed capabilities

Increases in embedded compute capacity at lower power are necessary to support systems of this type, with localized MI capability (at least at the level of feature comparison) also being useful, in both supporting the recognition algorithm and allowing operation with more basic camera equipment.

Containerization and self-configuration are required to be implemented at a higher level to allow easy installation. Containerization will allow

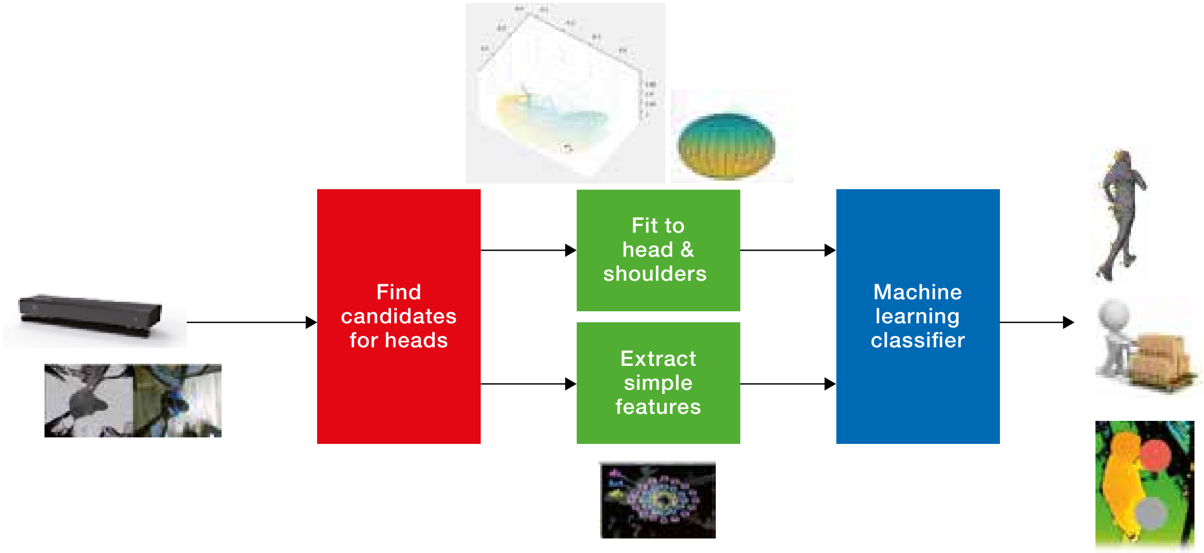


Figure A-9 | Tailgating use case

software to be delivered in discrete, secure, bounded packages, where the API connections can be tightly defined via microservices, without any attendant library or development language/OS requirements. Containerization will allow multiple IoT applications from different vendors to coexist on processing hardware and to be maintained and upgraded separately with no reference to applications of other vendors or effect on their operation, while allowing information sharing along pre-defined interfaces, E/W-bound at the device or gateway level.

Bandwidth increases within both public and private radio networks (see 5GPP) as well as capability to hand back and forth seamlessly between these networks will assist in decoupling cameras and processing nodes.

A.6.4.4 Necessary future Standards

Addition of Standards governing the encoding of feature data in embedded processors [59] would allow greater portability of feature data between different processor types without limited manufacturing freedom to innovate.

Self-configuration will be facilitated by a detailed set of recommendations/protocols for discovery, advertisement of device capabilities and registration on the network. Some of these could be based on, or related to, the emergent 5GPP Standards in this area.

Containerization of IoT applications (particularly at the gateway level) would be greatly facilitated by the creation of a common Standard for virtualization support on IoT nodes. This would be an extension of the ground covered by OCI [60] which has started a general effort.

A Standard governing E/W data sharing at this level between applications using a microservices Standard may be useful.

A.7 Fire detection via surveillance cameras

A.7.1 Scope

Potential areas of application include all indoor areas: warehouses, banks, offices, hospitals etc.

A.7.2 Objectives

Currently the level of damage caused by a fire has an inverse relationship to the time taken to detect it. Objective: detect it sooner – less damage.

A.7.3 Narrative of the use case

Using an MI module with access to video streams incoming from non-specialized video cameras, scan those streams using a trained model for traces of a fire. Cameras are linked to a processing module located either on-device, in-premise or in the cloud. Latency and bandwidth in this case are both issues in an in-premise or cloud solution.

A.7.3.1 Complete description

- Name of the use case: video detection of fire using an existing video surveillance network
- Requirements for bandwidth: high (approx. 20 MB/second, low latency <5 ms, contention an issue) between camera and processing node (if not co-located); or low (1 kB/day if the processing is local).
- It is worth noting here that depending on where the "intelligence" decides that the level of bandwidth the network is required to bear lies, there is a 10^5 scale difference between input and output bandwidth.

A.7.3.2 Diagram of use case

See Figure A-10

A.7.4 Use case conditions

A.7.4.1 State of the art

Currently no broad-based solutions to this problem are widely deployed.

A.7.4.2 Technology gaps

While this can be achieved currently, both cameras and compute nodes are needed to operate on low power equipment with more basic cameras. Coupled with this issue, it is challenging to install solutions due to the need for calibration to an environment.

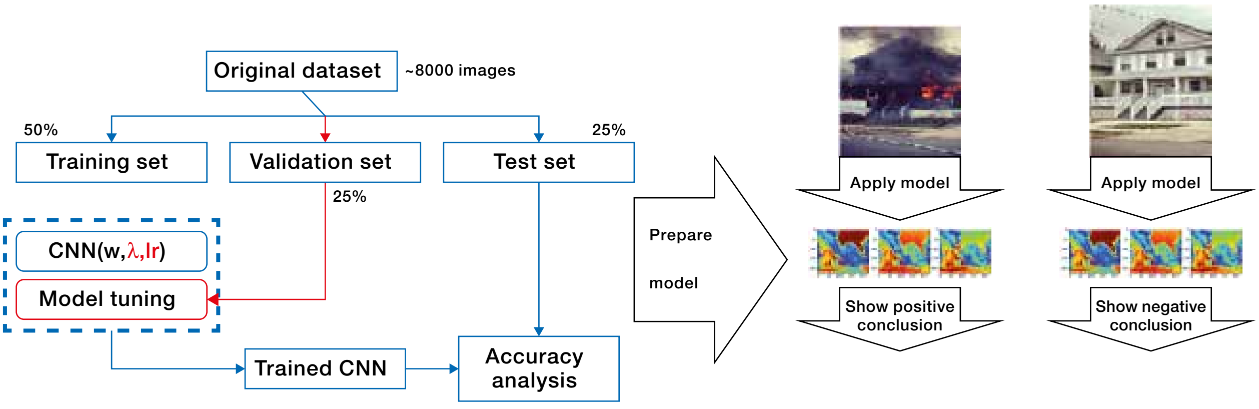


Figure A-10 | Fire detection use case

A.7.4.3 Needed capabilities

Increases in embedded compute capacity at lower power are necessary to support systems of this type, with localized MI capability (at least at the level of feature comparison) also being useful, in both supporting the recognition algorithm and allowing operation with more basic camera equipment.

Containerization and self-configuration are required to be implemented at a higher level to allow easy installation.

A.7.4.4 Necessary future Standards

Addition of Standards governing the encoding of feature data in embedded processors [59] would allow greater portability of feature data between different processor types without limited manufacturing freedom to innovate.

Self-configuration will be facilitated by a detailed set of recommendations/protocols for discovery, advertisement of device capabilities and registration on the network. Some of these could be based on, or related to, the emergent 5GPP Standards in this area.

Containerization of IoT applications (particularly at the gateway level) would be greatly facilitated by the creation of a common Standard for virtualization support on IoT nodes. This would be an expansion on the ground covered by OCI [60] which has started a general effort in this direction.

A Standard governing E/W data sharing at this level between applications using a microservices Standard may be useful.

Bibliography

[1] Ericsson, *Hyperscale cloud – reimagining data centres from hardware to applications*, May 2016 [Online]. Available: http://www.ericsson.com/res/docs/whitepapers/wp-hyperscale-cloud.pdf. [Accessed 19 September 2017].

[2] SATYANARAYANAN, M., *The Emergence of Edge Computing*, IEE Computer, Vol. 50, pp. 30–39, January 2017.

[3] International Data Corporation, *IDC FutureScape: Worldwide Internet of Things 2017 Predictions*, November 2016 [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=US40755816. [Accessed 19 September 2017].

[4] SIMSEK, M. et al., *5G-Enabled Tactile Internet*, IEEE Journal on Selected Areas in Communications, Vol. 34 (No. 3), March 2016.

[5] ITU-T, *The Tactile Internet*, International Telecommunication Union, August 2014.

[6] IEC, *Factory of the Future,* White Paper, 2017 [Online]. Available: http://www.iec.ch/whitepaper/futurefactory. [Accessed 19 September 2017].

[7] Mitsubishi Electric, *e-F@ctory*, 2017 [Online]. Available: http://sg.mitsubishielectric.com/fa/en/download_files/solutions/e_Factory.pdf. [Accessed 19 September 2017].

[8] PLAN.ONE, *PLAT.One Platform*, 2017 [Online]. Available: https://www.sap.com/products/iot-platform-cloud.html. [Accessed 19 September 2017].

[9] 5G-PPP, *5G Automotive Vision,* October 2015 [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-White-Paper-on-Automotive-Vertical-Sectors.pdf. [Accessed 19 September 2017].

[10] European Commission, *European strategy on Cooperative Intelligent Transport Systems (C-ITS),* 30 November 2016 [Online]. Available: https://ec.europa.eu/transport/themes/its/c-its_en. [Accessed 19 September 2017].

[11] European Commission, *5G for Europe: An Action Plan*, 14 September 2016 [Online]. Available: ec.europa.eu/newsroom/dae/document.cfm?doc_id=17131. [Accessed 19 September 2017].

[12] IEEE 802.11p-2010, *IEEE Standard for Information technology – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 6: Wireless Access in Vehicular Environments.* 15 July 2010 [Online]. Available: https://standards.ieee.org/findstds/standard/802.11p-2010.html. [Accessed 19 September 2017].

[13] DOKIC, J., MÜLLER, B., MEYER, G., *European Roadmap: Smart Systems for Automated Driving,* European Technology Platform on Smart Systems Integration (EPoSS), April 2015, [Online]. Available: http://www.smart-systems-integration.org/public/documents/publications/EPoSS%20Roadmap_Smart%20Systems%20for%20Automated%20Driving_V2_April%202015.pdf. [Accessed 19 September 2017].

[14] *Dedicated Short-Range Communications (DSRC) Fact Sheet,* Intelligent Transport Systems Joint Program Office, U.S. Department of Transportation [Online]. Available: http://www.its.dot.gov/factsheets/pdf/JPO-034_DSRC.pdf. [Accessed 19 September 2017].

[15] 5G Americas, *V2X Cellular Solutions*, October 2016 [Online]. Available: http://www.5gamericas.org/files/2914/7769/1296/5GA_V2X_Report_FINAL_for_upload.pdf. [Accessed 19 September 2017].

[16] BEDO, J-S., CALVANESE STRINATI, E., CASTELLVI, S., CHERIF, T., FRASCOLLA, V., HAERICK, W., KORTHALS, I., LAZARO, O., SUTEDJO, E., USATORRE, L., WOLLSCHLAEGER, M., *5G and the Factories of the Future*. 5G-PPP White Paper, 2015 [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-White-Paper-on-Factories-of-the-Future-Vertical-Sector.pdf. [Accessed 19 September 2017].

[17] KOTT, A., SWAMI, A., WEST, B. J., *The Internet of Battle Things*, IEEE Computer, vol. 49, p. 70-75, December 2016.

[18] *Proximity-based services (ProSe); Stage 2,* TR 23.303 3GPP; December 2016 [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=840. [Accessed 19 September 2017].

[19] GOODYEAR, M., LOUIS, J.H., *Defining the Security Domain,* University of Kansas, 2015 [Online]. Available: http://slideplayer.com/slide/2353814. [Accessed 19 September 2017].

[20] 5G-Ensure, *5G Security Architecture* [Online]. Available: http://www.5gensure.eu/5g-ensure-architecture. [Accessed 19 September 2017].

[21] Global Platform, *Internet of Things White Paper*, May 2014 [Online]. Available: https://www.globalplatform.org/documents/whitepapers/IoT_public_whitepaper_v1.0.pdf. [Accessed 19 September 2017].

[22] ANCUTA CORICI, A., EMMELMANN, M., LUO, J., SHRESTHA, R., CORICI, M., MAGEDANZ, T., *IoT inter-security domain trust transfer and service dispatch solution*, 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), December 2016.

[23] TAPSCOTT, D., TAPSCOTT, A., *Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World*, Portfolio, Penguin Random House, New York, 2016.

[24] SWAN, M., *Blockchain: Blueprint for a new economy*, O'Reilly Media, Sebastopol, California, 2015.

[25] SZABO, N., *The Idea of Smart Contracts*, [Online]. Available: http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/idea.html. [Accessed 19 September 2017].

[26] SWANSON, T., C*onsensus-as-a-service: a brief report on the emergence of permissioned, distributed ledger systems*, 6 April 2015 [Online]. Available: http://www.ofnumbers.com/wp-content/uploads/2015/04/Permissioned-distributed-ledgers.pdf. [Accessed 19 September 2017].

[27] ANTONOPOULOS, A.M., *Mastering Bitcoin: Unlocking Digital Cryptocurrencies*, O'Reilly Media, Sebastopol, California, 2015.

[28] SIGNORIN, M., *Towards an internet of trust: issues and solutions for identification and authentication in the internet of things*, University Pompeu Fabra, Barcelona, Spain, 2015.

[29] IBM, *Empowering the edge: Practical insights on a decentralized Internet of Thing*s. April 2015 [Online]. Available: https://www-935.ibm.com/services/multimedia /GBE03662USEN.pdf. [Accessed 19 September 2017].

[30] GOTTHOLD, K., ECKERT, D., *Deutschland erkennt Bitcoin als "privates Geld*. Welt N24 16, August 2013 [Online]. Available: https://www.welt.de/finanzen/geldanlage/ article119086297/Deutschland-erkennt-Bitcoin-als-privates-Geld-an.html. [Accessed 19 September 2017].

[31] PETERS, M., *Software-Defined Storage: A Buzzword Worth Examining*, 18 January 2013 NetworkComputing.com [Online]. Available: http://www.networkcomputing.com/ storage/software-defined-storage-buzzword-worth-examining/1334995080. [Accessed 19 September 2017].

[32] Available: https://www.edgexfoundry.org

[33] oneM2M, TS-0001-V2.10.0, *Functional Architecture,* August 2016 [Online]. Available: http://www.onem2m.org/images/files/deliverables/Release2/TS-0001-%20Functional_Architecture-V2_10_0.pdf. [Accessed 19 September 2017].

[34] Industrial Internet Consortium, *Industrial Internet Reference Architecture*, Version 1.7 June 2015, Report No. IIC:PUB:G1:V1.07:PB:20150601, [Online]. Available: https://www.iiconsortium.org/IIRA-1-7-ajs.pdf. [Accessed 19 September 2017].

[35] *OPC Unified Architecture Specification – Part 1: Overview and Concepts*. OPC Foundation, March 2015.

[36] 3GPP, 3GPP TS 29.273: *Evolved Packet System (EPS); 3GPP EPS AAA interfaces* [Online]. Available: www.3gpp.org. [Accessed 19 September 2017]

[37] PFAFF, B., LANTZ, B., HELLER, B., BARKER, C., COHN, D., TALAYCO, D., ERICKSON, D., CRABBE, E., GIBB, G., APPENZELLER, G., TOURRILHES, J., PETTIT, J., YAP, K.K., POUTIEVSKI, L., CASADO, M., TAKAHASHI, M., KOBAYASHI, M., McKEOWN, N., BALLAND, P., RAMANATHAN, R., PRICE, R., SHERWOOD, R., DAS, S., YABE, T., YIAKOUMIS, Y., LAJOS KIS, Z., *OpenFlow Switch Specification,* Version 1.1.0, February 2011 [Online]. Available: http://archive.openflow.org/documents/openflow-spec-v1.1.0.pdf. [Accessed 19 September 2017].

[38] Fraunhofer FOKUS, *Open5GCore: The Next Mobile Core Network Testbed Platform* [Online]. Available: www.open5gcore.org. [Accessed 19 September 2017].

[39] ETSI GS NFV-INF 001 V1.1.1, *Network Functions Virtualisation (NFV); Infrastructure Overview,* January 2015 [Online]. Available: http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/001/01.01.01_60/gs_NFV-INF001v010101p.pdf. [Accessed 19 September 2017].

[40] OpenStack, *Open source software for creating private and public clouds*, [Online]. Available: https://www.openstack.org. [Accessed 19 September 2017].

[41] Spirent, *NFV Validation across Boundaries,* Spirent White Paper, 2017 [Online]. Available: http://www.spirent.cn/-/media/White-Papers/Broadband/PAB/NFV_Validation_across_Boundaries_whitepaper.pdf. [Accessed 19 September 2017].

[42] ETSI, GS NFV-SEC 002 V1.1.1, *Network Functions Virtualisation (NFV); NFV Security; Cataloguing security features in management software,* August 2015. [Online]. Available: http://www.etsi.org/deliver/etsi_gs/NFV-SEC/001_099/002/01.01.01_60/gs_nfv-sec002v010101p.pdf. [Accessed 19 September 2017].

[43] FOKUS F., Open-source MANO implementation [Online]. Available: openbaton.github.io. [Accessed 19 September 2017].

[44] ETSI, GS MEC 003 V1.1.1, *Mobile Edge Computing (MEC); Framework and Reference Architecture*, March 2016 [Online]. Available: http://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf. [Accessed 19 September 2017].

[45] ETSI, *Mobile Edge Computing – Introductory Technical White Paper,* September 2014 [Online]. Available: https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf. [Accessed 19 September 2017].

[46] ETSI, *Multi-access Edge Computing,* 2016 [Online]. Available: http://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing. [Accessed 19 September 2017].

[47] GSMA, *Remote Provisioning Architecture for Embedded UICC,* May 2016 [Online]. Available: http://www.gsma.com/newsroom/wp-content/uploads//SGP.02_v3.1.pdf. [Accessed 19 September 2017].

[48] 3GPP, TR 23.799, *Study On Architecture For Next Generation System,* December 2016 [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3008. [Accessed 19 September 2017].

[49] NGMN Alliance, *NGMN 5G White Paper,* 2015 [Online]. Available: https://www.ngmn.org/uploads/media/NGMN_5G_White_Paper_V1_0.pdf. [Accessed 19 September 2017]

[50] 3GPP, 3GPP Radio Characteristics for IoT, 2016 [Online]. Available: http://www.3gpp.org/images/articleimages/iot_large.jpg.

[51] SIGFOX, *The world's first cellular operator dedicated to M2M and IoT*, [Online]. Available: http://www.iotglobalnetwork.com/public/files/product/1aeb87ba26088c15a4a707656fcce9f2.pdf. [Accessed 19 September 2017].

[52] SIGFOX, *Sigfox UNB technology vs. other cellular technologies*, [Online]. Available: http://www.iotglobalnetwork.com/public/files/product/0d8208975415b38ac7321ed8b4ac537a.pdf. [Accessed 19 September 2017].

[53] LinkLabs, *SigFox vs. LoRa – A comparison between technologies & business models*, [Online]. Available: https://www.link-labs.com/sigfox-vs-lora. [Accessed 19 September 2017].

[54] LoRa Alliance, *LoRa Alliance Technology*, [Online]. Available: https://www.lora-alliance.org/What-Is-LoRa/Technology. [Accessed 19 September 2017].

[55] Fraunhofer FOKUS, Open5GMTC, toolkit for M2M connectivity, [Online]. Available: www.open5gmtc.org. [Accessed 19 September 2017].

[56] 3GPP, TR 23.83, *Local IP Access and Selected IP Traffic Offload (LIPA-SIPTO),* 2011 [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=909. [Accessed 19 September 2017].

[57] Mitsubishi Electric, *e-Factory,* November 2016 [Online]. Available: http://app.mitsubishielectric.com/app/fa/download/search.do?kisyu=/sol/efactory&mode=catalog. [Accessed 19 September 2017].

[58] IndexMundi, *World Energy Consumption*, [Online]. Available: http://www.indexmundi.com/g/g.aspx?c=xx&v=81. [Accessed 19 September 2017].

[59] Intel, *Intel IoT Platform*, [Online]. Available: http://www.intel.com/content/www/us/en/internet-of-things/infographics/iot-platform-infographic.html. [Accessed 19 September 2017].

[60] Open Container Initiative, [Online]. Available: https://www.opencontainers.org/about. [Accessed 19 September 2017].

[61] LoRa Alliance, [Online]. Available: https://www.lora-alliance.org/what-is-lora. [Accessed 19 September 2017].

[62] NarrowBand IoT, [Online]. Available: https://en.wikipedia.org/wiki/NarrowBand_IOT. [Accessed 19 September 2017].

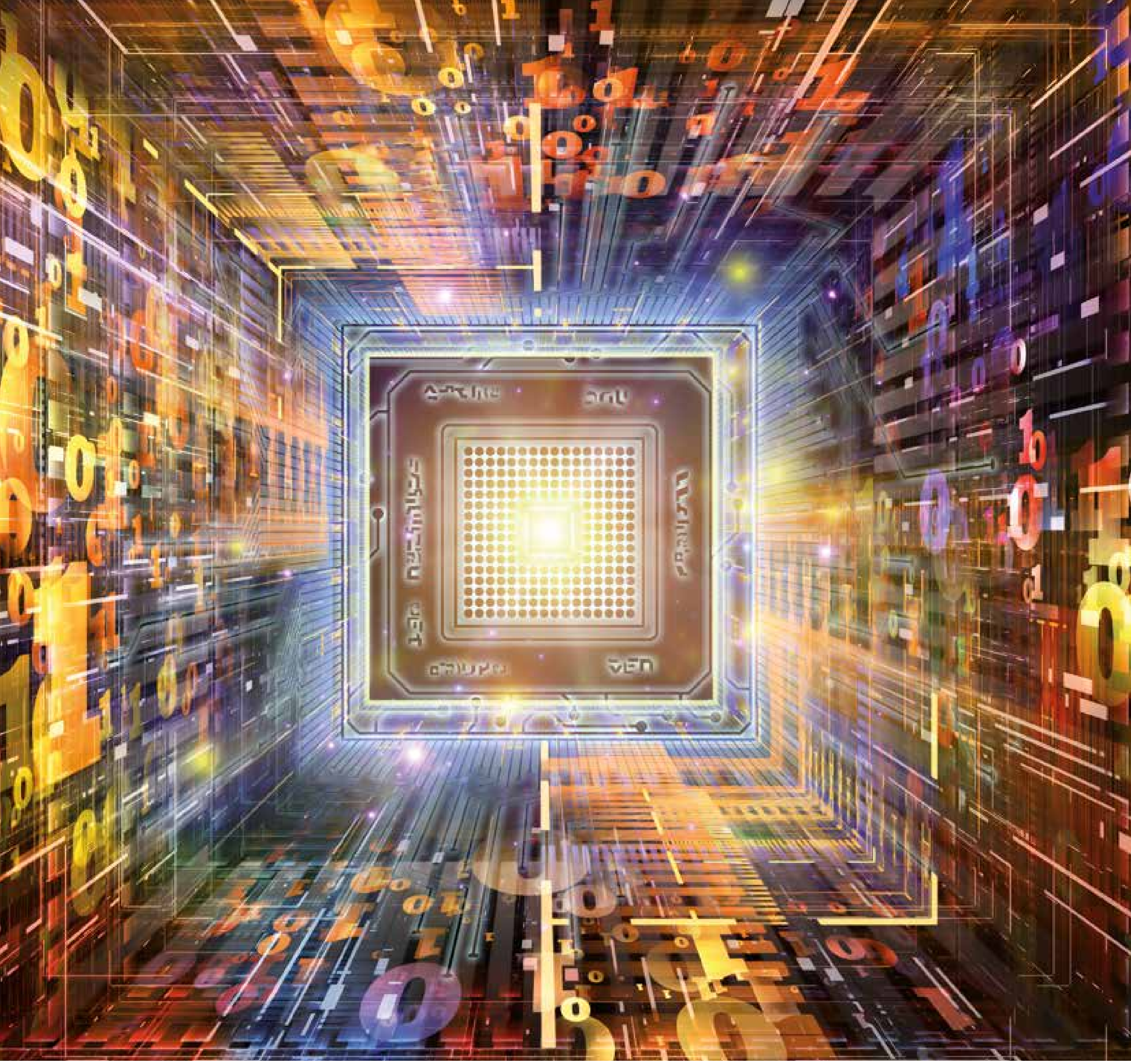
[63] IEC 62443-3-3:2013, *Industrial communication networks – Network and system security – Part 3-3: System security requirements and security levels*

[64] Zigbee technology, [Online]. Available: https://en.wikipedia.org/wiki/ZigBee. [Accessed 19 September 2017].

[65] ZWave Technology, [Online]. Available: https://en.wikipedia.org/wiki/Z-Wave. [Accessed 19 September 2017].

[66] ResinOS, [Online]. Available: https://resinos.io. [Accessed 19 September 2017].

Notes



ISBN 978-2-8322-4854-6

CHF 50.-

International
Electrotechnical
Commission ®

3 rue de Varembé
PO Box 131
CH-1211 Geneva 20
Switzerland

T +41 22 919 02 11
info@iec.ch
www.iec.ch

IEC WP Edge intelligence:2017-10(en)