E4S – Specifications Summary

EDGE for SMART

SECONDARY SUBSTATION SYSTEMS

1. Acronyms and Abbreviations

AdvLVS	Advance Low Voltage Supervision
DC/DCU	Data Concentrator Unit
DERMS	Distributed Energy Resource Management
DMS	Distribution Management System
E4S	Edge for Smart Secondary Substation
FTP	File Transfer Protocol
HES	Head End System
IOM	IO Modules
ICM	Inner Communication Model
I/O	Input/Output
IP	Internet Protocol
IT	Information Technologies
K8s	Kubernetes
LV	Low Voltage
LVS	Low Voltage Supervision
MDM	Meter Data Management
MQTT	Message Queuing Telemetry Transport
MV	Medium Voltage
OMS	Outage Management System
OT	Operational Technologies
PCM	Primary Compute Module
PLC	Power Line Communications
PM	Power Module
PRIME	Powerline Intelligent Metering Evolution
RESTful	Web Services that conform to Representational State Transfer
RTU	Remote Terminal Unit
SCADA	Supervisory Control And Data Acquisition
SAML	Security Assertion Markup Language
SS	Secondary Substation

SSP	Secondary Substation Platform
SCM	Secondary Compute Module
SM	Switch Module
SMI	System Management Interface
STG-DC	Sistema de Telegestión (Central system). Also known as AMM
TBC	To be Clarified
TBD	To be Defined
URI	Uniform Resource Identifier
WS	Web Services
WAN	Wide Area Network
WOT-A	Web of Things Architecture

2. Document Scope

The Edge for Smart Secondary Substation System (E4S) is designed to enable the digitalized grid. To improve the local decisions in substations, more information is required on each device electrically connected to the grid. A highly integrated secondary substation platform is required to collect and store data, providing virtual space for running multiple applications. A virtualized platform to consolidate functions and to offer a converged infrastructure is envisioned.

This document is a summary of the E4S full specifications and provides a general overview of the full specification. This summary covers the Edge Applications and the System Architecture of the E4S Specifications. The full specification will be available for public release soon and will be available here - https://www.e4salliance.com

2.1 Stakeholders

2.2 Edge Applications

The following functions summarize some of the common applications that could be implemented as part of the Edge computing transformation. A brief description is presented for some of these functions and the expected input and outputs, especially for those requiring a specific I/O card.

2.2.1 Data Concentrator for smart metering

This application is responsible for gathering smart meter data, e.g., through PLC PRIME, and send it back to the MDM/HES through the WAN connections available in the Data Concentrator.

In the case of PRIME DCUs, the protocol used downlink over PLC is DLMS/COSEM, while the protocol used to uplink to the backend systems is STG-DC, which is based on web services (SOAP implementation used) and FTPs.

I/O: The application needs a gateway between the application and the meters, e.g., PRIME Base Node connected to the three phases (and neutral) of the LV network in the secondary substations. This connection is used for the PLC injection.

2.2.2 Advanced Low Voltage Supervisor (AdvLVS)

This application is responsible of monitoring the low voltage panels in the secondary substation. It communicates DLMS over RS-485 with the I/O cards installed in the LV panel and STG-SABT protocol, based in web services (SOAP implementation used) with the backend systems.

I/O: The application requires the installation of I/O cards in each phase of the LV panel. These I/O cards can measure instantaneous values (V, I, P and Q) of each phase.

2.2.3 Grid Topology Identification

This application calculates the low voltage topology, identifying line and phase for each smart meter connected to the LV panel. It uses the information from the Data Concentrator and Advanced Low Voltage Supervisor.

I/O: The application does not require any additional I/O because it uses data from other applications.

2.2.4 Outage Identification

This application aims to identify outages in the LV network, estimating the size of the outage and the impact in the customers. It uses the information from Data Concentrator, Advanced Low Voltage Supervisor and Grid Topology applications.

I/O: The application does not require any additional I/O because it uses data from other applications.

2.2.5 Transformer Regulation Monitoring & Control

This application monitors LV parameters of the transformer and regulates output voltage to adapt the power quality to the status of the grid. It uses information from Data Concentrator, Advanced Low Voltage Supervisor and Grid Topology applications. It provides an interface with the transformer controller in order to modify transformer output. It is used when an OLTC transformer is installed in the secondary substation.

I/O: The application requires an I/O module that can operate the transformer controller.

2.2.6 Video Analysis for Visual Monitoring

This application can cover multiple use cases based on the video signal coming from one or several digital cameras taping different areas of the substation.

I/O: The application requires one or several video cameras.

Some potential use cases are the following:

- Oil levels detection and losses identification.
- SF6 levels detection and losses identification.
- Non authorized access identification
- Health and safety verification during work execution
- Fire identification and prediction

2.2.7 Substation Load Balancing & Load Profiling

This application calculates load statistics of the secondary substation, per phase, circuit, and transformer in order to suggest a better distribution of customers.

I/O: The application does not require any additional I/O because it uses data from other applications.

2.2.8 Demand and Generation Prediction Analytics

This application calculates demand and generation predictions based on historical data of smart meters and low voltage panels.

I/O: The application does not require any additional I/O because it uses data from other applications.

2.2.9 Distributed Energy Resource Management (DERMS)

This application operates distributed energy resources to respond to specific grid situations. Due to regulation, it is still unclear how to manage resources, but the benefit to the grid is certain.

I/O: The application requires an I/O to communicate with the DER.

2.2.10 Advanced Medium Voltage Monitoring and Control

This application is similar to a traditional RTU. However, it is not yet defined how it will operate or interface with the MV assets.

I/O: The application requires an I/O to communicate with the RTU.

3. System Architecture

3.1 Current Automation in Utilities

The efficiency and reliability potential of connecting field devices in the distribution grid has moved utilities in the last years to massive deployment to automate their operations.

The most common applications include distribution automation applications, such as supervisory control and data acquisition (SCADA), Automated Meter Infrastructure (AMI), outage management system (OMS), and distribution management system (DMS) to name a few.

To avoid scalability and interoperability problems, and to secure the investments, a layered architecture of communication and electrical equipment has been deployed trying to comply with the following principles:

- The use of open standards and technologies (IEC, IEEE, ITU-T) to avoid vendor lock-in
- International practices and homologation procedures for equipment/protocols validation
- Integration of the latest security mechanism and standards
- Layer network and Fault-tolerant communication systems
- Change in processes and operational behaviors

While significant, this centric architecture faces the exponential growth of distributed energy generation, electric vehicles, and a myriad of more sophisticated applications. Besides adapting to the new energy services, regulators and customers demand a better quality of the service, the network's resiliency, and an extended portfolio of services.

Integrating all this functionality demands a more dynamic and distributed approach where decision and control become more agile and closer to the last mile. A new paradigm is required, where the primary and secondary substations will take a more proactive role in operating and deploying new applications on the go.

3.2 State of the Art on Distributed Computing Platforms

The paradigm change experienced in data architectures across industries is the shift from cloud computing to edge computing. Edge computing is the new model of distributed processing architecture for distributed networks. In contrast to the strongly consolidated cloud computing, Edge computing brings information processing closer to the network's edges, reducing latency and increasing the system response speed. This model responds to a market's needs in which the amount of information from the end systems is continuously growing.

In the field of power distribution, there is an increasing sensorization and monitoring at all levels of the grid, which increases the amount of information available by several orders of magnitude. The availability of more information and greater computing capacities allows the deployment of more advanced control algorithms, which in some cases, are deployed in distributed platforms. These systems result from the convergence of 3 groups of technologies:

1. **IoT (Internet of Things)**: Enabling the management of millions of devices and their information flows in a scalable and cost-effective manner.

2. **Real-time Operation (OT)**. Providing integration capabilities and direct control of electrical equipment and energy management devices in real-time.
3. **Virtualization and CaaS management (Container as a Service**), enabling:
    - Efficient encapsulation of management, analysis, and control functions.
    - An open and safe ecosystem on which to extend functionality.
    - Centralized management, maintenance, and deployment platform for millions of distributed computing nodes.

Distributed computing architectures have different advantages for the management of distributed infrastructures, such as distribution networks:

- **Minimize reaction times**. Centralized information management requires time to receive information and act. Distributed control makes it possible to respond to use cases that require much shorter analysis and response times by being closer to the events.
- The distribution of processing, analysis, and reaction capacity **maximizes the value of the information received while minimizing the volume of information** on which the operator can and must react.
- **Optimizes the use of existing infrastructure** (communications, databases, and processing) and minimizes the overall cost of operation and maintenance:

The above benefits tradeoff with:

- The extra cost related to the increase in deployment points.
- The need for more processing and computing equipment
- Reshape of the cybersecurity concept.

Distributed computing has evolved in the last years from fog computing to the current concept of edge computing. Although conceptually similar, there are practical differences in the reach of the computing distribution:

- **Fog computing** uses a centralized system that interacts with industrial gateways and embedded computer systems on a local area network.
- **Edge computing** performs much of the processing on embedded computing platforms directly interfacing to sensors and controllers.

The following figures show the comparison between the Edge and Cloud paradigms.

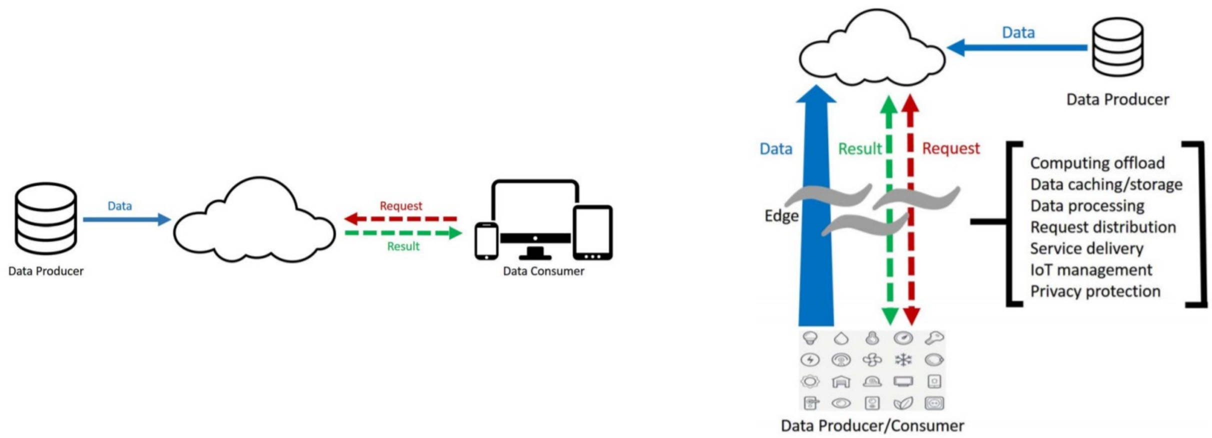

Figure 1 - Edge and Cloud paradigms (Shi, 2016)

Industrial applications can be classified into two broad types of Edge deployments depending on the way the Edge nodes organize their workload:

1. A single Edge layer is deployed in the field, and each Edge node is responsible for processing local information and coordinate with the cloud layer.
2. Edge nodes create local computing clouds that share local processing loads and may be organized in different tiers depending on their computing capacity.

3.3 Edge Computing

Edge computing is associated with the evolution of computing, networking, and storage architectures near the endpoints, sensors, machines, mobile devices at the edge of the network.

The Cloud and Datacenter Infrastructure with appropriate adaptation are attractive technologies to address the needs of Edge compute Infrastructure, in particular in technologies of:

- Virtualization and Containerization
- Software-Defined Networking
- Software-Defined Storage
- Resource management, automation, and orchestration.
- Streaming Data Analytics

With its adoption of the most modern concepts developed in the IT domain, and at the same time with its need to satisfy the requirements of the Operational domains (OT), such as deterministic behaviors in networking, computing and storage, sensor and actuator support, and aggregation, and sometimes even safety support, Edge computing is the perfect conduit for the highly promising convergence of IT and OT in many key Industrial IoT verticals.

With the need for dynamic and distributed Intelligence at the Edge for electricity generation and distribution, the infrastructure at the Edge has to provide the ability for applications to run with the following requirements:

- Accelerate Application Innovation & Delivery

For faster innovation and delivery of applications developed by multiple subject matter experts (e.g., DCU, LVS, Outage, Topology), Applications need to abstract from the hardware and platform. The application needs to be portable and fully composable (specify all the requirements for run time environment in the specification template). The platform would read the required specification for the application and create the desired run time environment (CPU, storage, networking, security). To achieve these requirements, its necessary to build

a) Platform as Virtualized platform, such that it's abstracted from the hardware. In case of hardware failure, or better hardware available, the platform should migrate to new hardware with ease and stateful migration.
b) Containerized applications, portable as a standalone or group of applications.
c) Ability to deliver and upgrade single or a collection of applications to one substation or group of sub-stations from the centralized management station.

- Real-Time Response

Real-time determinism for applications such as circuit-breakers, overcurrent protection, and outage monitoring is met using dedicated purpose build hardware. With the push to bring cloud technologies such as virtualization and containerization to the edge for application consolidation, edge platforms should provide real-time execution. The virtualized environment must ensure the application can **run in a deterministic manner and provide a real-time response** for the events happening at the edge devices (e.g., meters, feeders, outage response).

- Autonomous Operation

Even with the WAN connectivity improvement over the years, it is not guaranteed that communication is always available 24x7 between all Edge nodes and a centralized management station. The main reason to push the intelligence to the edge is to allow sub-stations to make a faster, local, and autonomous decision, with or without involvement with the centralized DSO's OT/IT systems. Thus, it is required to support independent operations for the sub-stations. The substations should react to dynamic application needs and recovers to transient failures locally, including application failure, Edge node failure etc. The ability to support reconciliation with the DSO after the network connectivity restoration is also required.

- Edge Analytics for Data-Driven Insights

With rich data available on the Edge from meters and feeders, there is a requirement to mine this data for applications like Demand prediction, Generation prediction, and visual condition monitoring. Edge Infrastructure should be capable of connecting data producers (meters and feeders) to data consumer applications (outage, demand prediction etc.) using a standard pub/sub-broker. The Edge infrastructure should also provide the capability to store the data from the sensors in a time series Database and have rich visualization tools to customize the data view. Providing a data platform with capabilities to clean, transform and load (ETL) and run streaming analytics with simple and complex event processing with AI/ML engine on the Edge node would be very desirable, as next-generation sub-station becomes the center of intelligence to address the dynamic demand and supply of energy.

Edge computing should provide data extraction tools, data management, data storage, and data analytics at the Edge node itself. The substation Edge node should offer the following base functionalities:

- Rich set of data connectors for extracting data from sensors.
- Topic-Based Subscription

- Efficient time-series database.

Depending on the final use cases, the following functionalities can be desirable:

- Per node replicated (fault-tolerant) and distributed time series database.
- Graphical tool to visualize the time series database.
- Tools to build data pipelines for:
    - Data Cleaning and Schema Conversion
    - Statistical Analysis
    - Machine Learning
    - Anomaly Detection

- Scale

With the decentralized approach of Edge computing, the problem of scale is solved in a horizontal and distributed fashion. Edge nodes' ability to work autonomously and make the decision locally eliminates the need for big, centralized compute and big centralized data lakes. Just as built-in attributes of the distributed architecture, the Edge computing quickly adds thousands to tens of thousands of Edge clusters. This project requires the edge compute platform to support thousands of sub-stations distributed across various geographical locations.

- Security

With the Intelligence pushed into a secondary sub-station where the locations may not be physically secured, the utilities' perimeter security must be redefined. Traditional centralized entry point inspection firewalls can no more provide the necessary perimeter security and DMZ. The Intelligent Edge platform needs to address these expanded and distributed perimeter security risks. It also needs to act as the central pillar (root of trust) to provide security for the less intelligent connected devices (meters, feeders), provide OT/OT separation, OT/IT separation, and provide application-level security.

- Hardware

The E4S hardware platform will have common computing node or nodes and will have multiple application-specific input-output interfaces. The hardware system should be modular and extensible. It should utilize open-standard interfaces and common open protocols. The system should be manageable and have hardware-based root of trust capabilities.

3.4 E4S High-Level System Architecture

A high-level overview of the system architecture is shown below in Figure 5, with the different components that could be part of a deployed solution.

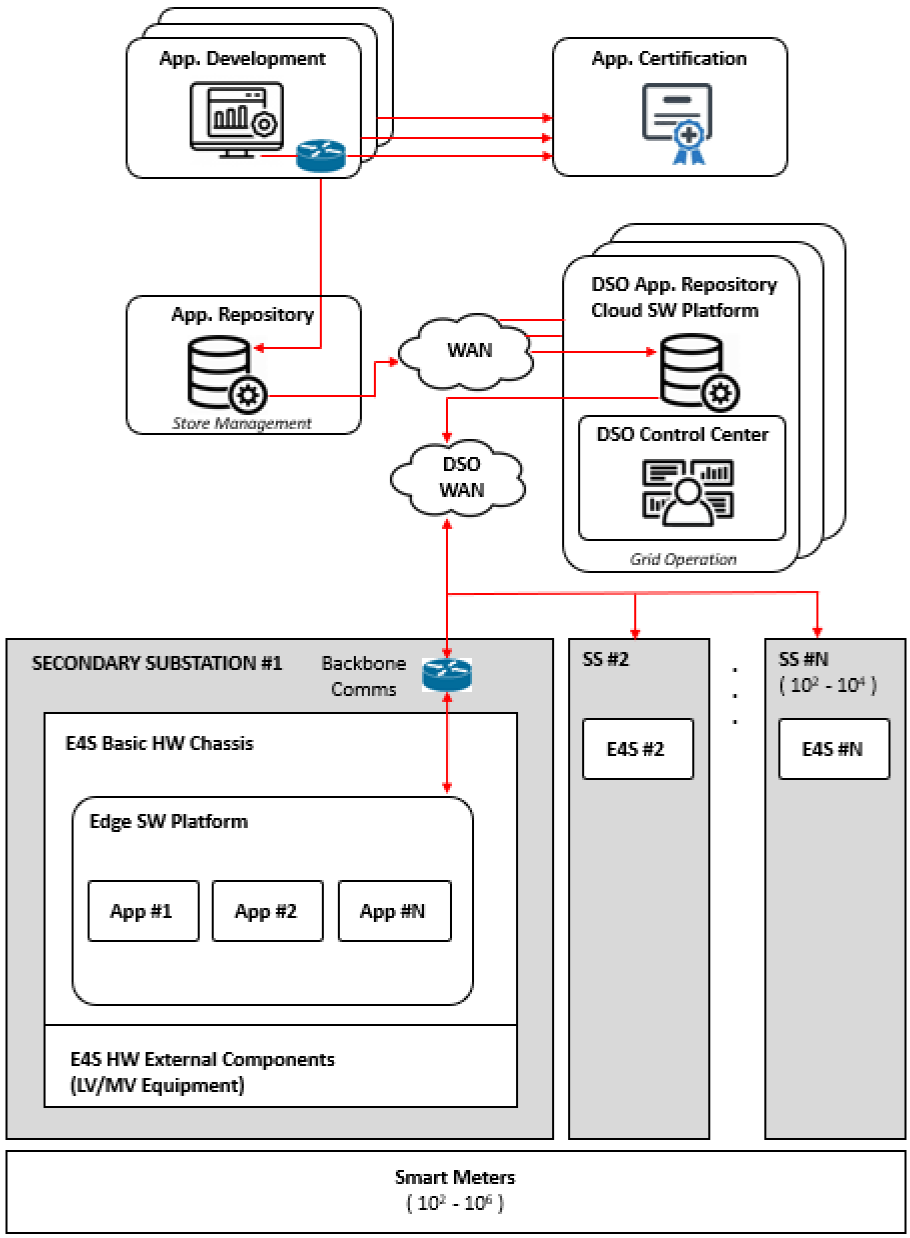

Figure 2 - High-Level Architecture

3.4.1 Operation and Application Management

The three actors that will interact and operate with the E4S system include:

1. **Asset Owner:**
    a. HW Administrator: IT HW expertise to manage resources of system and field devices.
    b. SW administrator: IT SW expertise. Manage networking and security aspects as well as application deployment and monitoring, both at the data center and field devices.
2. **Application owner**: each application will have an owner responsible for the functionality delivered by the app, including configuration and parametrization. It will interact with the asset owner to coordinate any action on the devices, besides only a single application.
3. **Field engineer**: Following indications from the asset owner and application owner, he will perform field activities at the E4S site.

3.4.2 Centralized Components at the DSO

In addition to the components deployed at the Edge, managing devices, and the grid edge domain, the E4S architecture requires a central layer (Cloud or On-Premise) capable of centrally and remotely deploying, managing, and monitoring thousands of Edge nodes deployed at the Edge of the grid. Two sets of central systems involved in the proposed architecture:

- **Business Systems**: The Edge nodes should enable the business data exchange between the devices and the business systems, supporting the communications protocols required for those systems to operate. Edge nodes and the Edge applications should enrich this data exchange through local data processing, local automation, and analytics. Business systems include but are not limited to SCADA, ADMS, MDMS, or OMS.
- **Central Edge Management system**: Responsible for the central monitoring and management of all the Edge nodes deployed in the field.

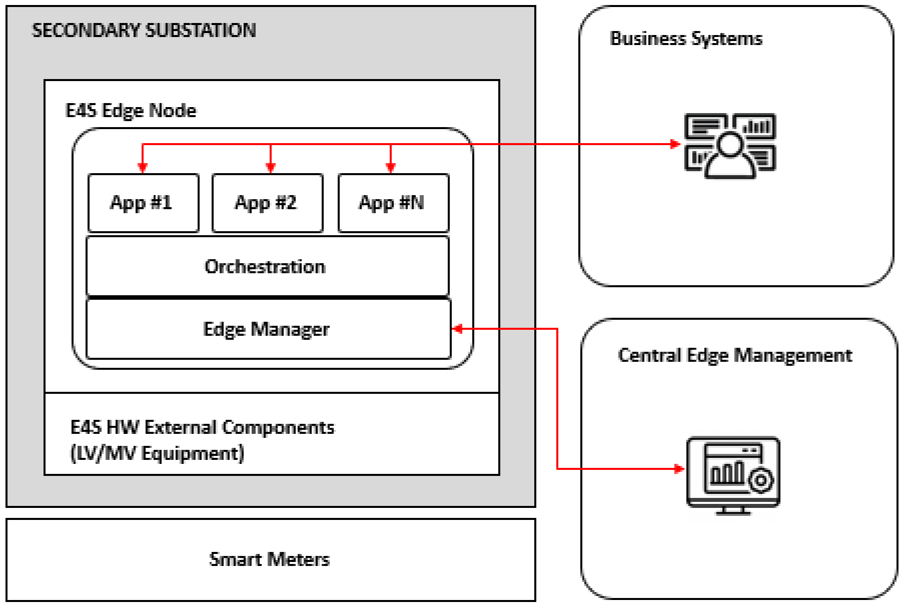

Figure 3 - E4S High-level systems access

E4S will focus on the Central Edge Management System specification, a new approach required to manage this massive deployment of distributed computing at the Edge of the grid.

The Edge Management System should provide the following functionality:

a) Complete inventory management of all deployed Edge nodes.
b) Grouped management of Edge nodes by secondary substation type.
c) Automatic registration of the devices managed by each of the Edge SS nodes.
d) Unified monitoring and management of the Edge nodes' operating status with an overview of the installed SS and details by installation: Use of resources, use of resources by deployed application, alarms and events.
e) Access to low-level management and administration tools for nodes and devices that enable it (SSH, SNMP, device WebUI, etc…).
f) Management and registration of deployed Edge applications.
g) DevOps environment for the deployment and verification of apps.
h) Massive management of centralized configurations and applications on Edge SS node families.
i) Automatic detection of anomalies in Edge nodes.
j) Support automated restart rules of Edge node and containers.
k) Secure communication and Edge management based robust encryption methods.

3.4.3 DevOps

The E4S Edge Management System must implement tools that aid and accelerate and optimize the process of development, validation, and deployment of new functionalities into the SS. To achieve this, the E4S Edge Management System must enable a DevOps environment with the following capabilities:

a) Continuous Integration and Delivery (CI/CD) system such as Git/Jenkins allows building/modifying and automatically testing new functionalities/projects for the SS, including its encapsulation in the form of containers.
b) Container repository.
c) The system must have a unified repository that manages all the versions of containers/virtual machines deployed in the SSs. This repository must track the different states of the containers: Obsolete, in tests, in production.
d) Configuration Repository (Infrastructure as Code) stores the basic configuration and system resource allocation parameters required for the various SS configurations. The repository must allow independent configuration profiles that can be applied to the different group types of SS.
e) Virtual SS test environment.

The system should be capable of generating an updated virtual replica of the different types of SS configurations.

These replicas must include the latest version of the containers and configurations of a typical SS environment as well as tools that allow similar entries in such a way as to verify the operation of new containers integrated into the E4S SS ecosystem.

3.5 E4S High-Level Edge Node Architecture

The distributed platform will provide the nodes with the layered components detailed in Figure 4. This section specifies the minimum infrastructure requirements for the edge. Common services are reviewed in Section 5 and details related to the applications in Section 6.

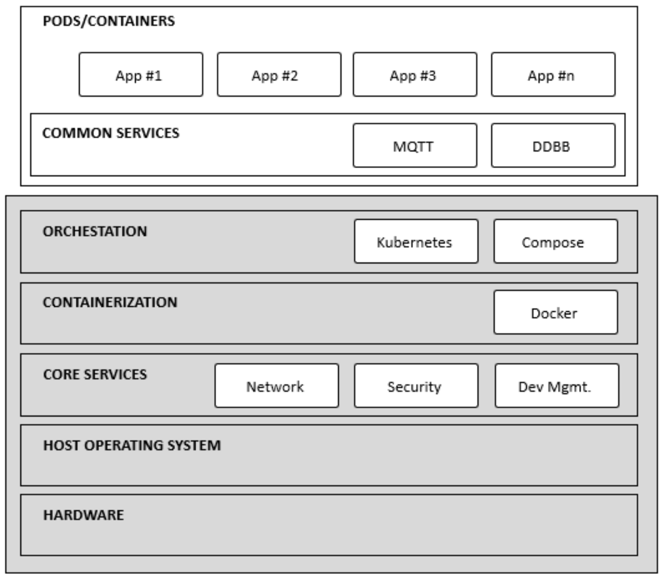

Figure 4 – Edge Node distributed components

3.5.1 Host Operating System

Requirement	At least one of these	Recommended
Operating System	A 64 bit Linux OS	Based on DSO requirements
Virtualization	Containers/Pods and VMs	Containers/Pods and VMs Application of additional containers isolation based on VMs, Kata Containers or VM over k8s virtualization capabilities.
Hypervisor	Type I (bare-metal or native) and Type II (hosted) hypervisors if needed. XEN, KVM, ACRN, vShpere.	KVM
Boot Loader	UEFI, BIOS	UEFI (to support secure boot)

Real-Time and Deterministic	1 ms-10ms or above deterministic workload guarantee	1 ms or above deterministic workload guarantee

3.5.2 Core Services

Requirement	At least one of these	Recommended
Security	SELinux, Seccomp, Apparmor, Container Security, Managed Firewalls (L3/L4)	Managed Firewalls, SELinux, Container Security policy.
Networking	Linux bridges, OVS, IOVisor, other CNIs	Linux bridges
Storage	Any format supported as Container Storage Interface (CSI). Ceph, Gluster,Local SSD, RAID, iSCSI, USB, NFS.	Local SSD with overlay2
Backup and Snapshot	Snapshot - live, snapshot- paused	Snapshot – live
Service Monitoring	Systemd restart, container restart	Systemd restart, container restart
Resource Monitoring	Built-in, 3rd party integrated	Built-in (e.g. Prometheus or Telegraf)

3.5.3 Containerization

Requirement	At least one of these	Recommended
Container Runtime	Dockers, Podman, Moby, LXC, CRI-O	Podman
Container images	Dockers images, OCI images	Docker images
Container deployment specification	Compose, Kubernetes Standard declarative way based on YAML or JSON and compliant at least with Compose or Kubernetes	Compose YAML

3.5.4 Orchestration

Requirement	At least one of these	Recommended
Orchestrator Architecture	K8s, K3s, Other light weight Edge orchestrators	Light weight edge orchestrator

Orchestration Master	Openshift, Tanzu, Rancher, Self-managed master	Self-managed (mature and more control to customize)