LF Edge Akraino Project Presentation

Ike Alisson
LF Edge Akraino TSC member &
LF Edge Akraino Documentation Sub-committee TSC Chair
November 24th, 2021

Table of Contents

AKRAINO

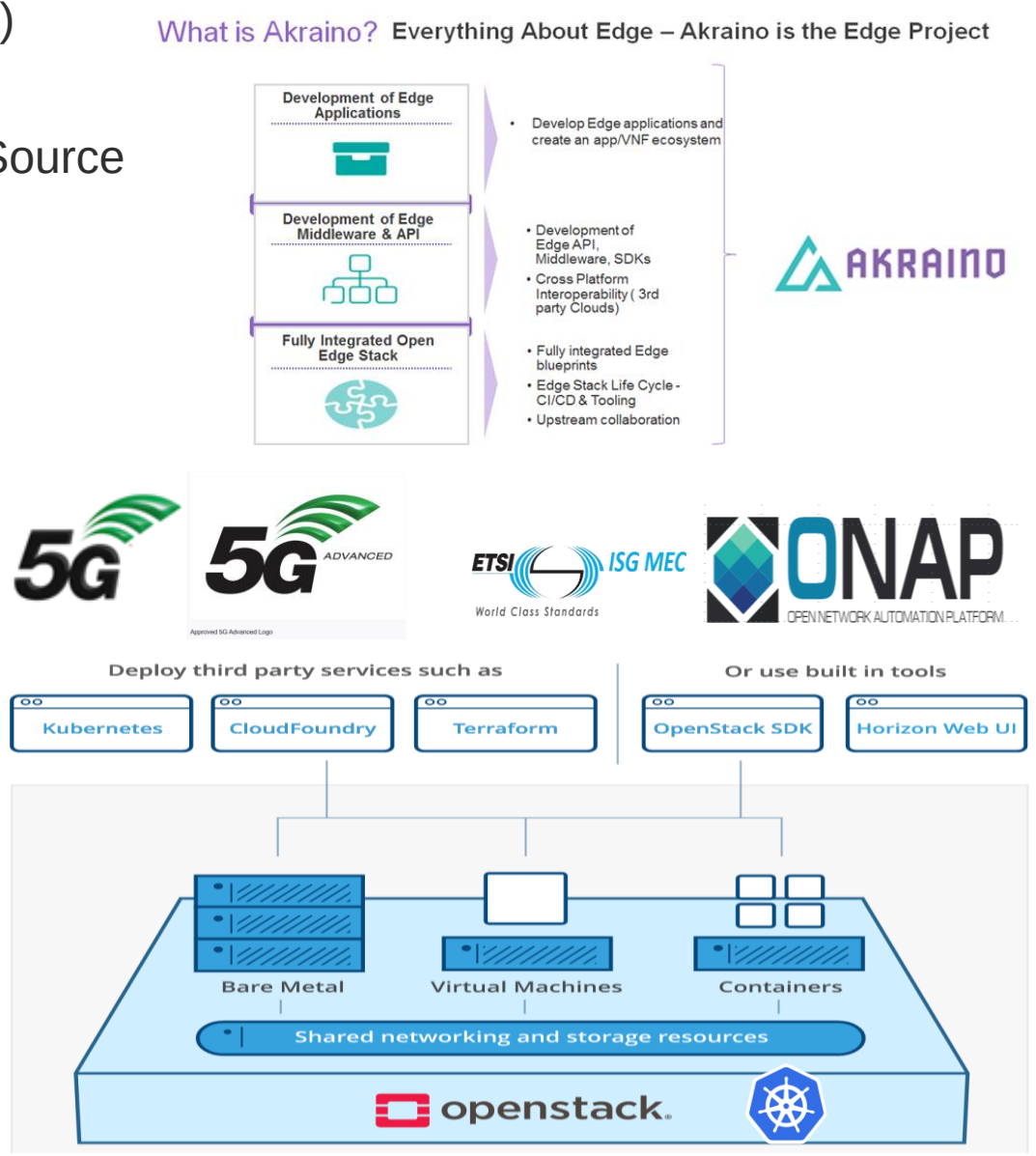
1.1. LF Edge Akraino Project Overview

- 20 < Blueprints (aka Integration Projects), BPs Proposals & Development Projects
  - set of Open Infrastructures & Application Blueprints (BPs)

- Coordination & Co-operation with Multiple Upstream Open Source Communities/SDOs as:
  - Airship,
  - LFN Anuket
  - OpenStack,
  - ONAP,
  - ETSI MEC,
  - GSMA,
  - TIP,
  - CNCF
  - O-RAN

Objective: To deliver a fully integrated stack

AKRAINO



1.1.1 LF Edge Akraino Project Overview: LF Edge TAC Stage 3 Project

- **LF Edge Stages - Definitions & Expectations**
  Every Foundation Project has an associated Maturity Level, as voted on under the approved Project Lifecycle Document (PLD) Process.

- Projects of all maturities have access to Foundation Resources.

- **Stage 3: Impact Stage ('Top-Level')** Definition
  - Projects that have reached their Growth Goals and are now on a Self-sustaining Cycle of Development, Maintenance, and Long-term Support.

Impact Stage projects are widely used in Production Environments and have Large, Well-established Project Communities with a number of Contributors from at least two (2) Organizations.

1.1.2 Akraino Project (Blueprint) Lifecycle States and Reviews phases

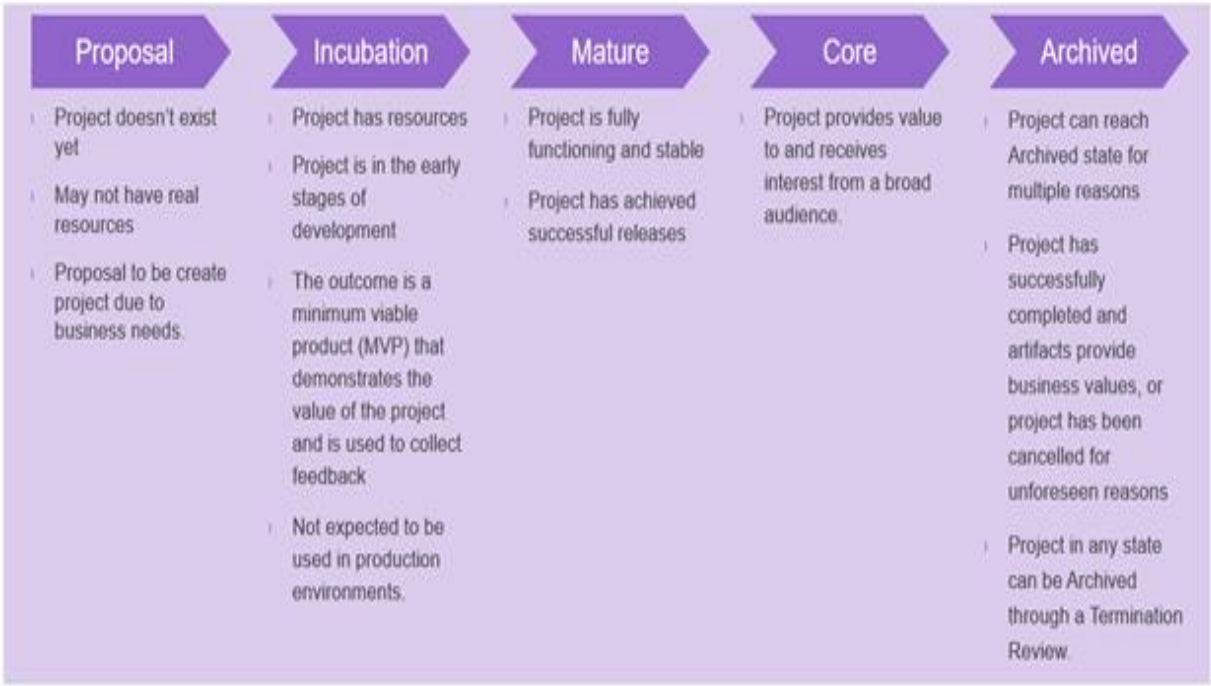
- **F**ive (5) states that Projects goes through.

- A Project Lifecycle may **extend across** Multiple Projects and Akraino Releases.

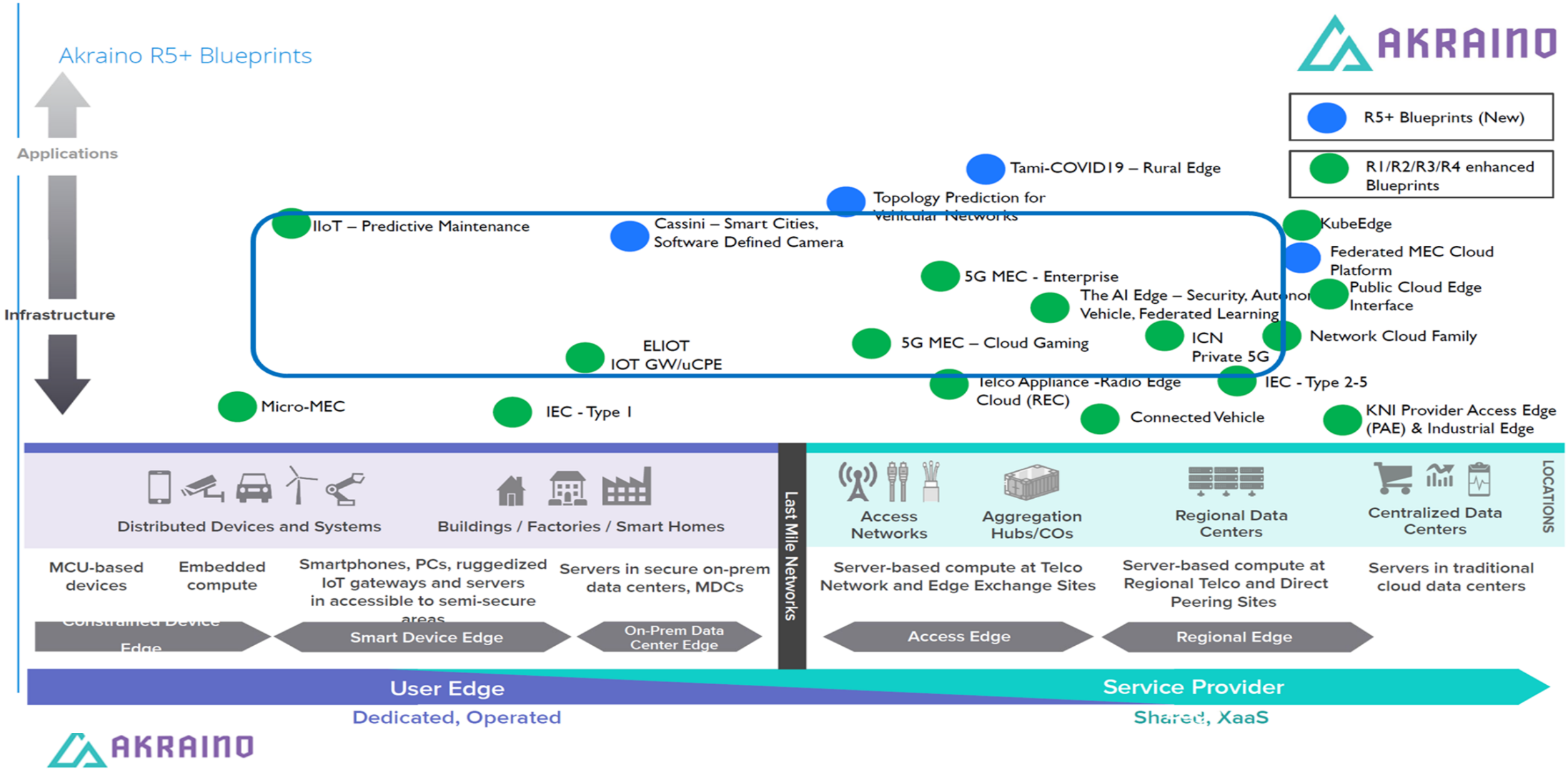
- The Procedure of moving from one(1) State to the next one is **independent from the Akraino Release Lifecycle** and the pace depends on each individual Project.

- In order to effectively review Project progress, **four (4) R**eviews are built-in to the Project Lifecycle, namely,

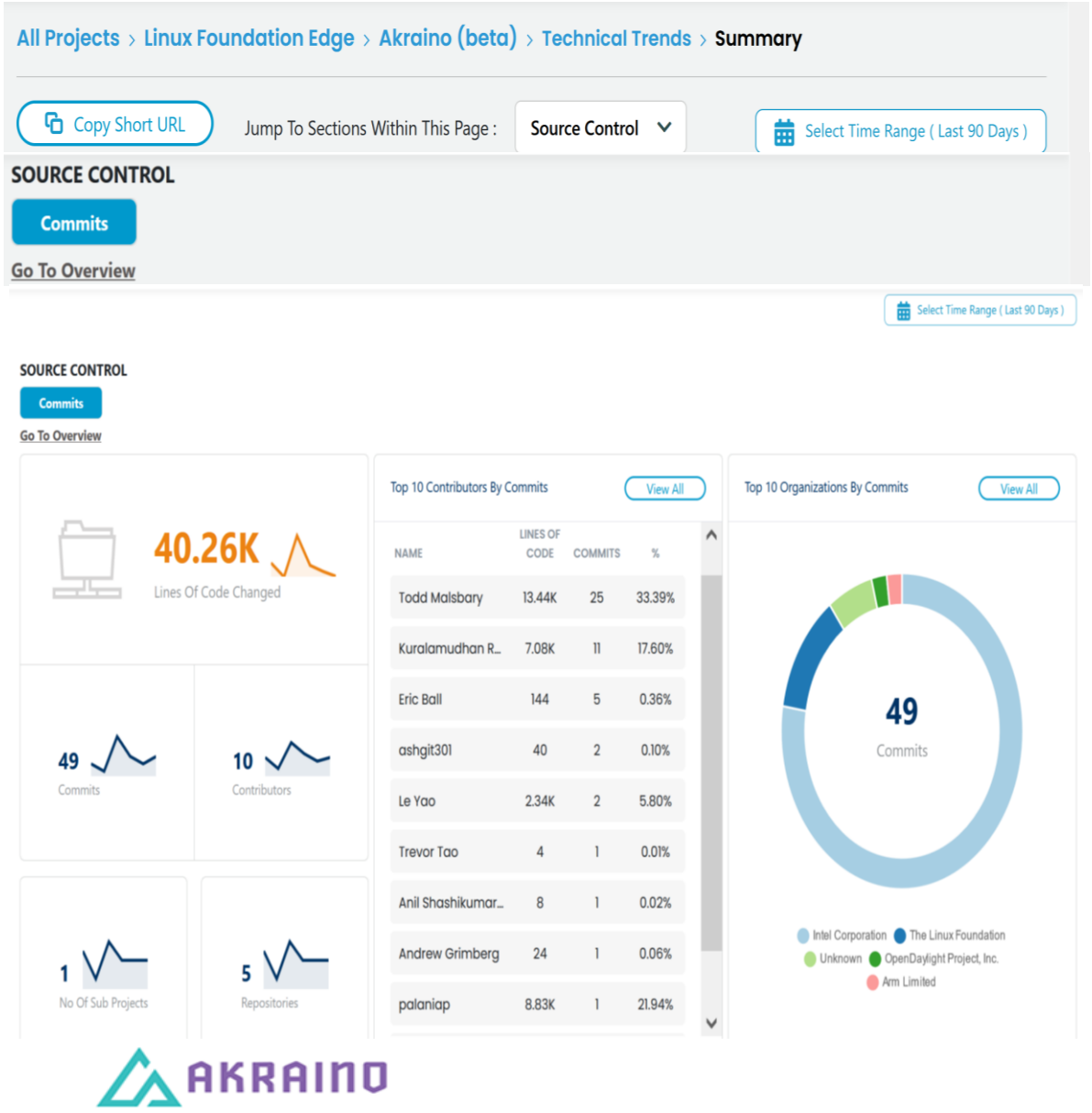
    1. Proposal,
    2. Incubation,
    3. Mature,
    4. Core
    5. Archived

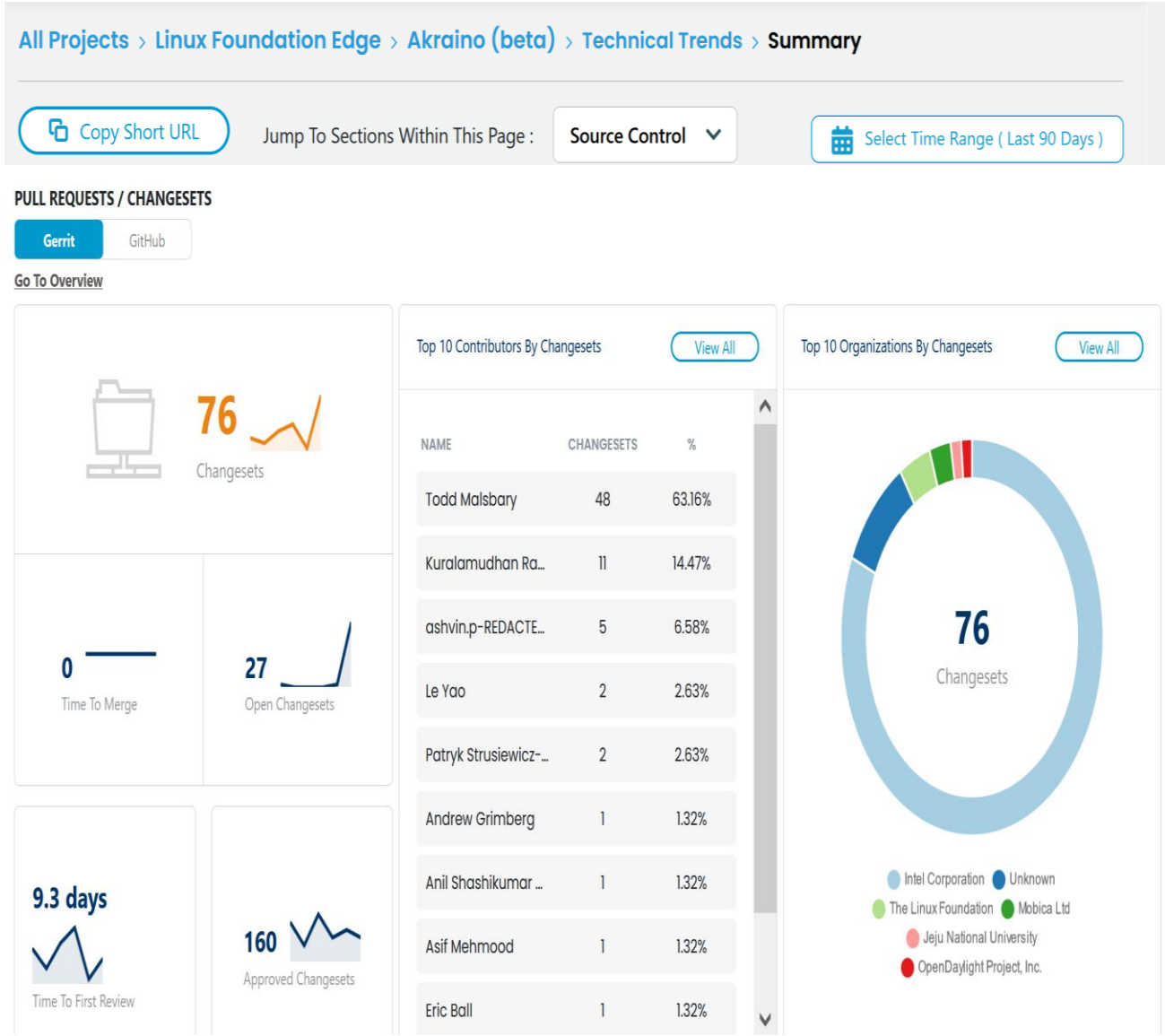


1.1.3 LF Edge Akraino Project Overview - R5 & R4 BPs

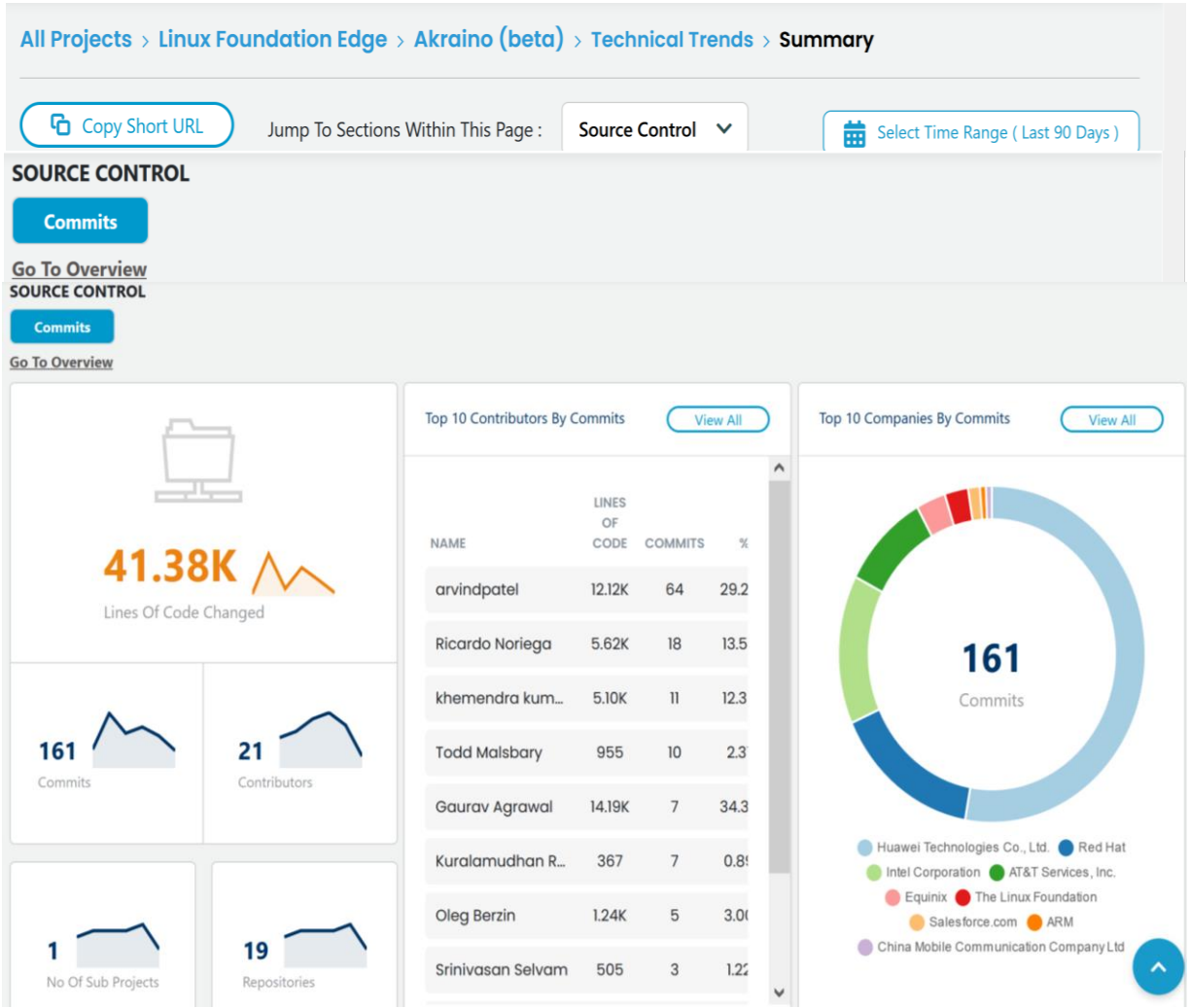


1.1. 4 LF Edge Akraino Project Analytics - Commits by Contributors and Companies - 1





1.1.4 LF Edge Akraino Project Analytics - Commits by Contributors and Companies - 2



1.1.5 Akraino Project TSC Sub-committees

Red Hat

The State of Enterprise Open Source

The State of Enterprise Open Source

A Red Hat® Report

In a stand-out finding of interest to our Telco Customers, 95% of respondents from the Telecommunications Industry report using Open Source.

The high-level takeaway of the report is that: "using Open Source SW across all Industries is no longer principally about making best use of IT Budgets.

Lower Cost of Ownership has fallen off the top spot and now sits in sixth (6th) position.

Today, the Strategic Benefits of using Open Source are valued more, including:

Top benefits of using enterprise open source

1. Higher quality software **35%**
2. Access to latest innovations **33%**
3. Better security **30%**
4. Ability to safely leverage open source technologies **30%**

AKRAINO

Akraino Edge Stack
Security Sub-Committee

September 24, 2021

Daniil Egranov
Security Sub-Committee Co-Chair, Akraino

Randy Stricklin
Security Sub-Committee Chair, Akraino

Akraino Security Team 2021 Accomplishments

- Automated Lynis, Vuls and Kube-Hunter Log Output Pass/Fail Analysis
- Lynis – Reviewed Required Tests
  - Formalized and Documented Lynis Incubation vs Maturity Requirements
- Platform Security for Akraino Blueprints
  - Arm
  - x86
- Release 4 and 5 Blueprint Reviews

AKRAINO

Akraino Security Team Future Plans

- Develop Minimum OS Version Support Document
  - Ubuntu, CentOS, RHEL CoreOS, Debian
- Develop Minimum Security Tool Version Support Document
  - Lynis, Vuls, Kube-Hunter, and OVAL (Vuls) database
- BluVal (Blueprint Validation):
  - Integrate Automated Lynis, Vuls and Kube-Hunter Pass/Fail
  - Enforce minimum versions of Vuls, Lynis and Kube-Hunter
- Version 1.0 Platform Security Whitepaper
- Investigate using LFX Security

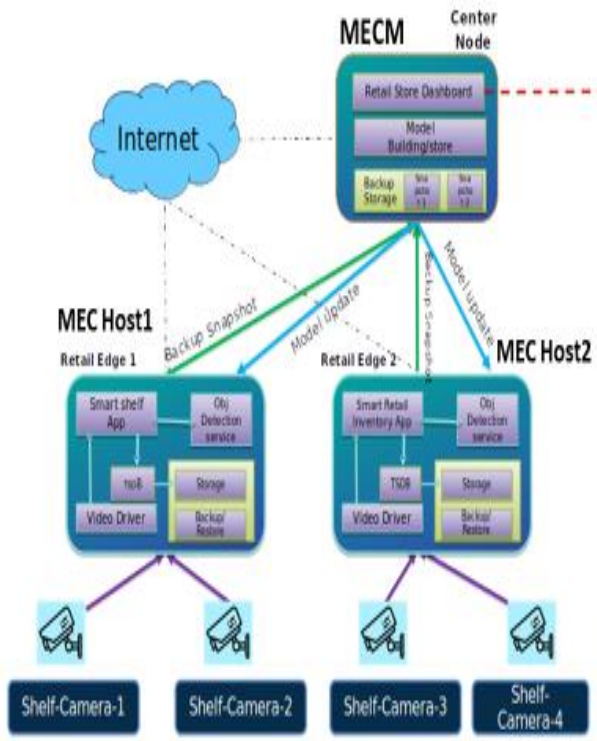
AKRAINO

ETSI MEC Update of the collaborations with OpenSource, with special focus on LF Edge Akraino

Presented by: **Jane Shen,**
VP of Technology Strategy, Mavenir
Technical Expert, ETSI MEC ISG

15.11.2021

Enterprise Application on Lightweight 5G Telco Edge Use Case



Demo Brief

Smart retail stores on Remote sites(ROBO).

Part 1: Smart shelf in retail store on lightweight ROBO edges.

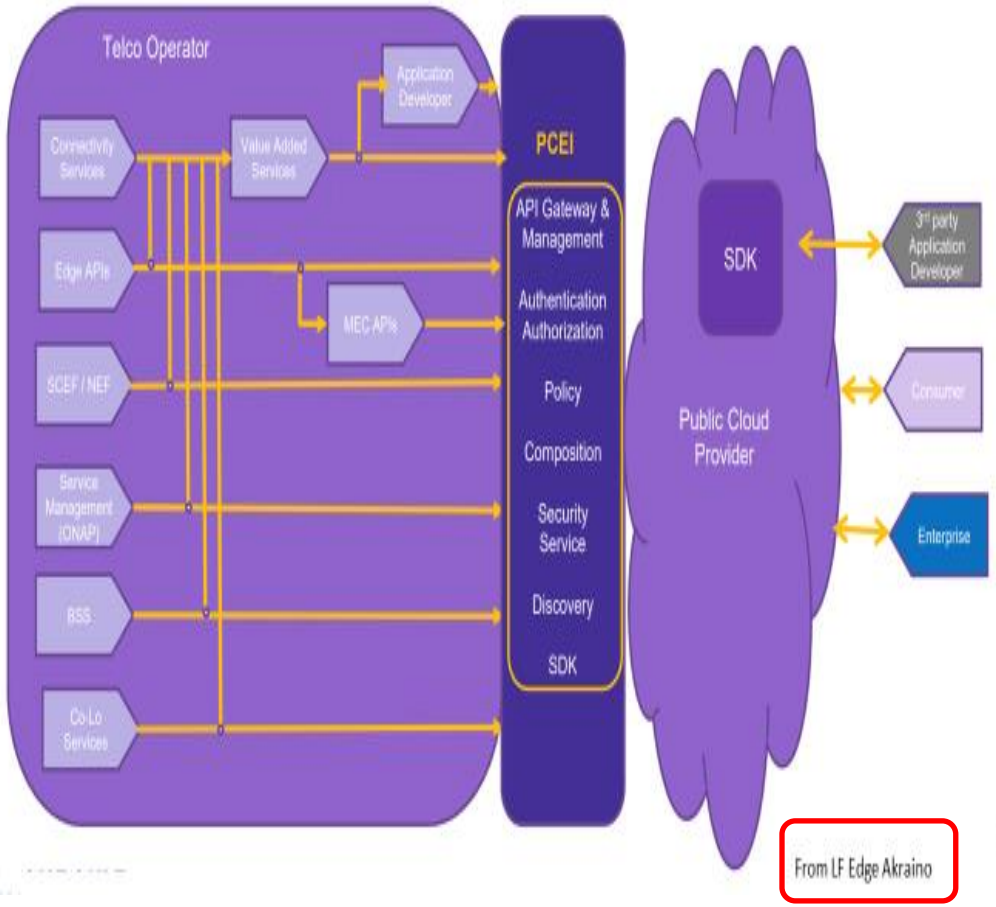
Part 2: In disaster scenarios, stateful Applications and data high availability with backup and restore of edge sites to remote cloud node.

1. Instantiate EALT Management and Edge Node. App Deployment on Edge Node(Smart retail App, Obj detection service, Portal)
2. Model download to Edges sites
3. Connect shelf cameras to Edge Node
4. Video driver stream to Smart retail application and object detection service to detect product and count.
5. Maintain Product inventory details in persistence TSDB which use platform storage service for persistence.
6. Backup the Applications and data and create disaster scenario.
7. Restore Application, state and its data from remote center site

From LF Edge Akraino

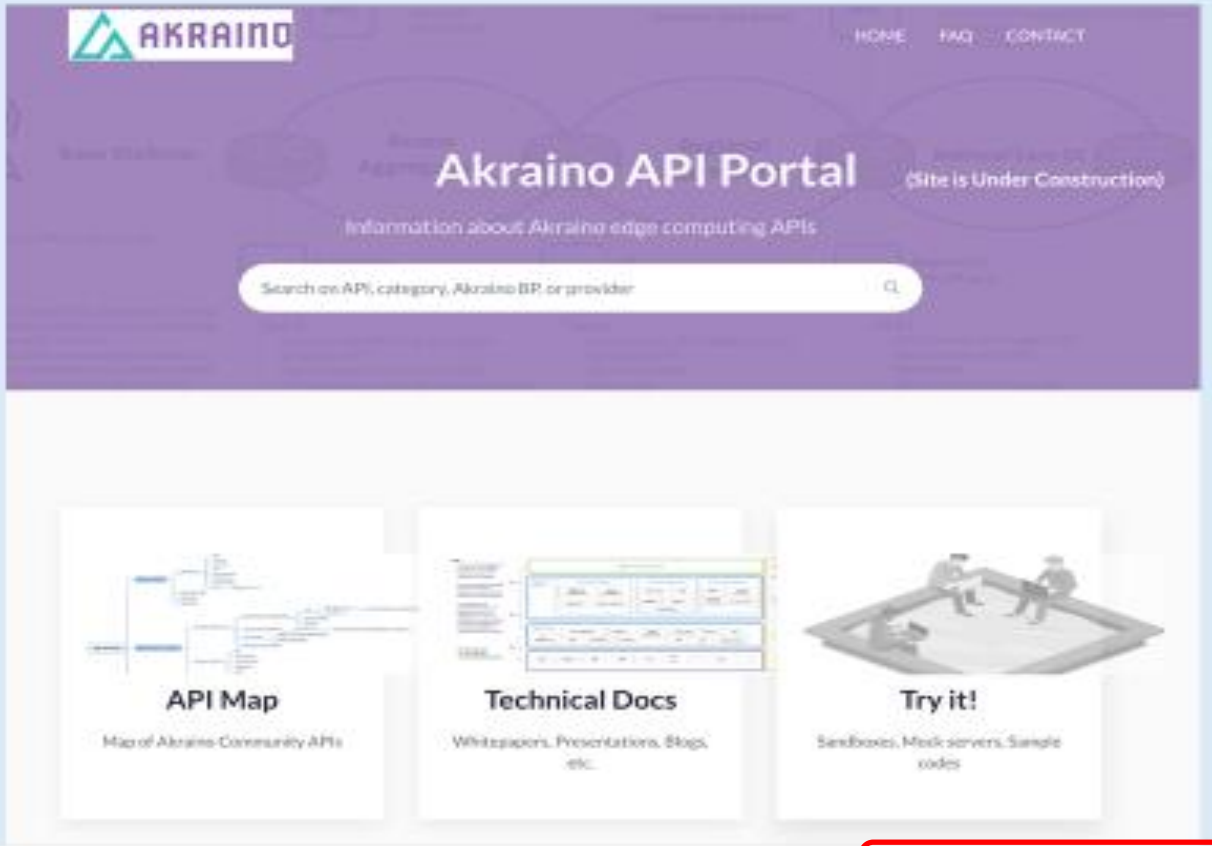
AKRAINO

Public Cloud Edge Interfacing (PCEI) Blueprint Projects



From LF Edge Akraino

An API Portal For Edge Developers



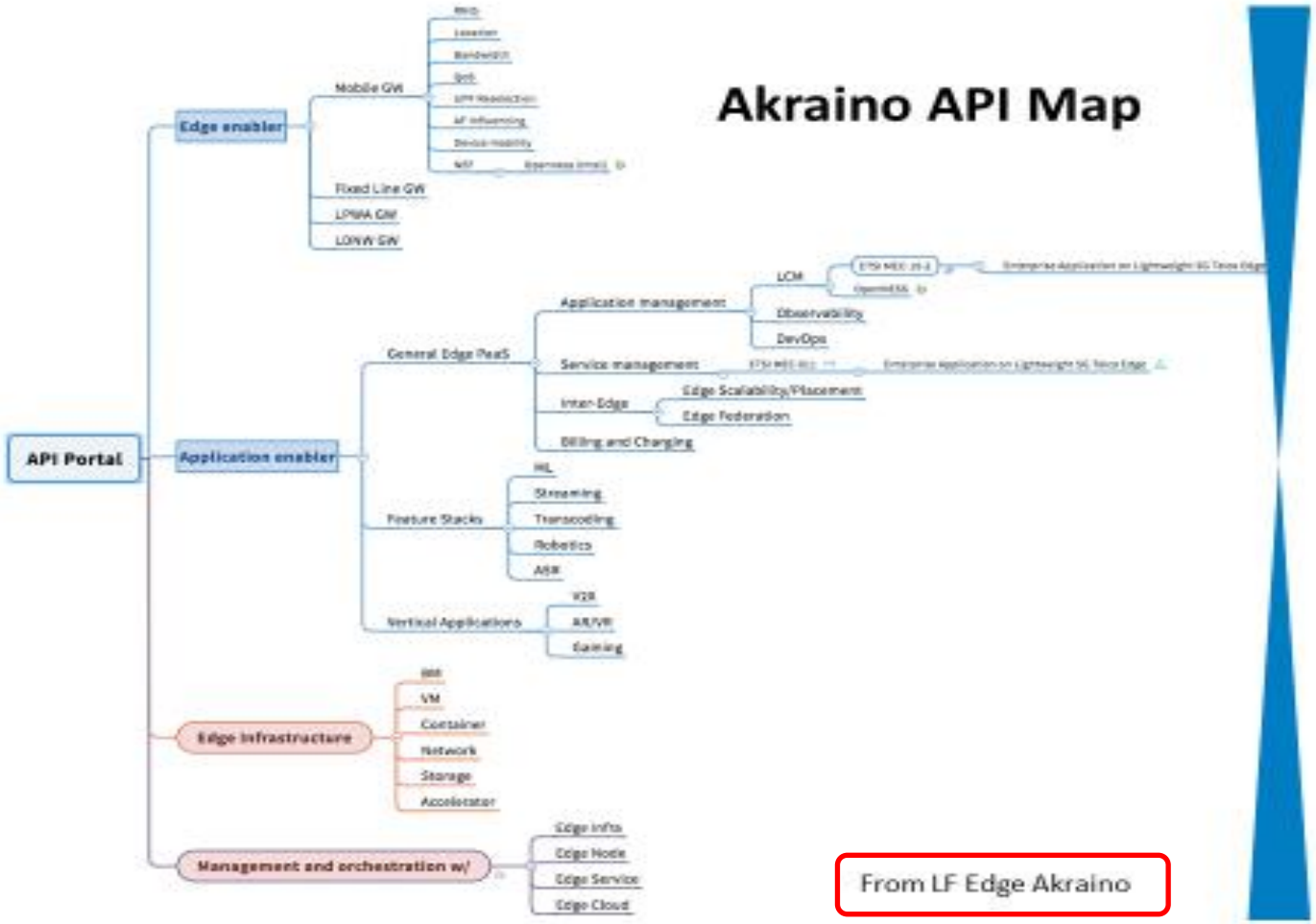
From LF Edge Akraino

- An API info hub of Akraino projects
- Cross reference with other relevant API information sites, e.g. ETSI MEC wiki, forge.etsi.org etc.
- Highlight API offerings from Akraino projects

https://apiportal.Akraino.org

10

An API Map For Akraino Project APIs



Akraino API Map

From LF Edge Akraino

ETSI MEC DECODE

Summarized by Walter Featherstone

Group Spec (GS)	GS Version	Service Name	apiName
MEC-010-2	2.1.1	Application lifecycle, rules & req management (Mm1 & Mm3)	app_pkgm app_lcm granting
	2.1.3		app_pkgm app_lcm granting
MEC-011	1.1.1	Platform Application Enablement (Mp1)	mp1
	2.1.1		mec_app_support mec_service_mgmt
	2.1.2		mec_app_support mec_service_mgmt
MEC-012	1.1.1	RNIS	rni
	2.1.1		rni
MEC-013	1.1.1	Location Service	location
	2.1.1		location
MEC-014	1.1.1	UE Identity Service	ui
MEC-015	1.1.1	BandWidth Management (BWM) service	bwm
	2.1.1	BandWidth Management (BWM) service and Multi-access Traffic Steering (MTS) service	bwm mts
MEC-016	1.1.1	UE App (Mx2)	mx2
	2.1.1		mx2
	2.2.1	Device App (Mx2)	dev_app
MEC-021	2.1.1	App Mobility Service (interface)	amsi
MEC-028	2.1.1	WLAN Access Information Service	wis
MEC-029	2.1.1	Fixed Access Information Service	fai
MEC-030	2.1.1	V2X Information Service	vis
MEC-033	2.0.2	IoT API	iot

12

IoT Platforms Competitive Landscape & Database 2020

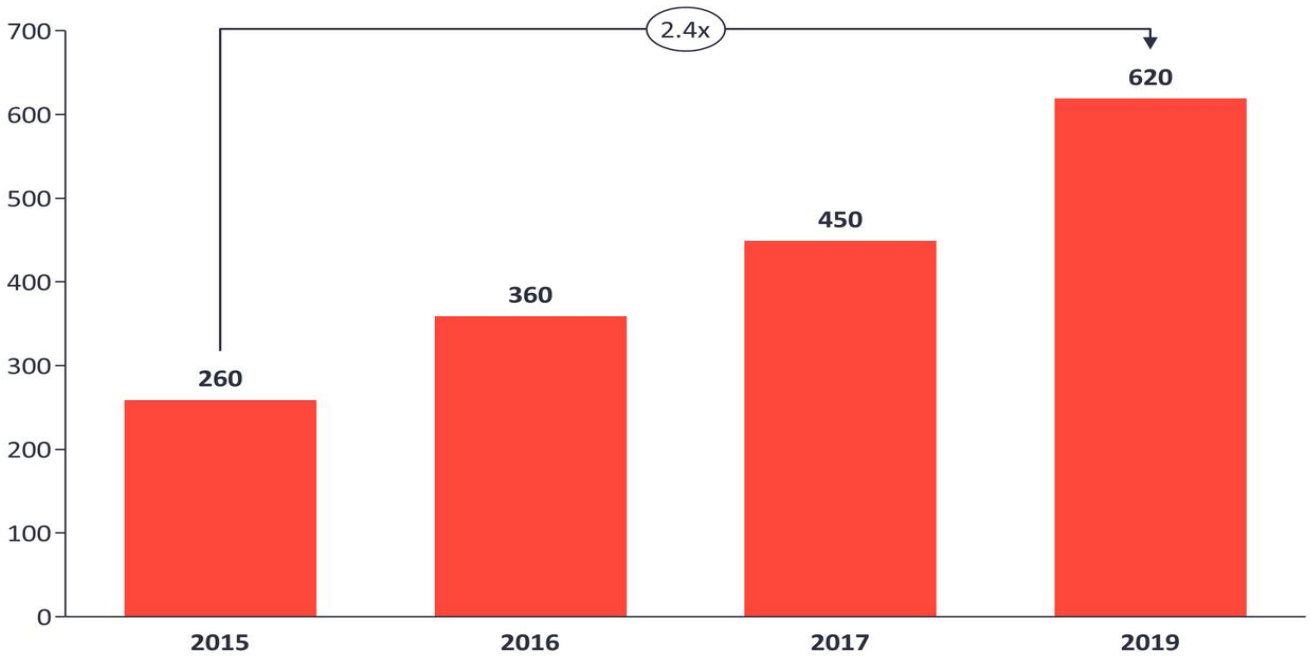
Database of 600+ IoT Platforms

IOT ANALYTICS
MARKET INSIGHTS FOR THE INTERNET OF THINGS

Insights that empower you to understand IoT markets

Number of publicly known "IoT Platforms" (2015-2019)

Number of publicly known "IoT Platforms" (IoT Analytics Research)



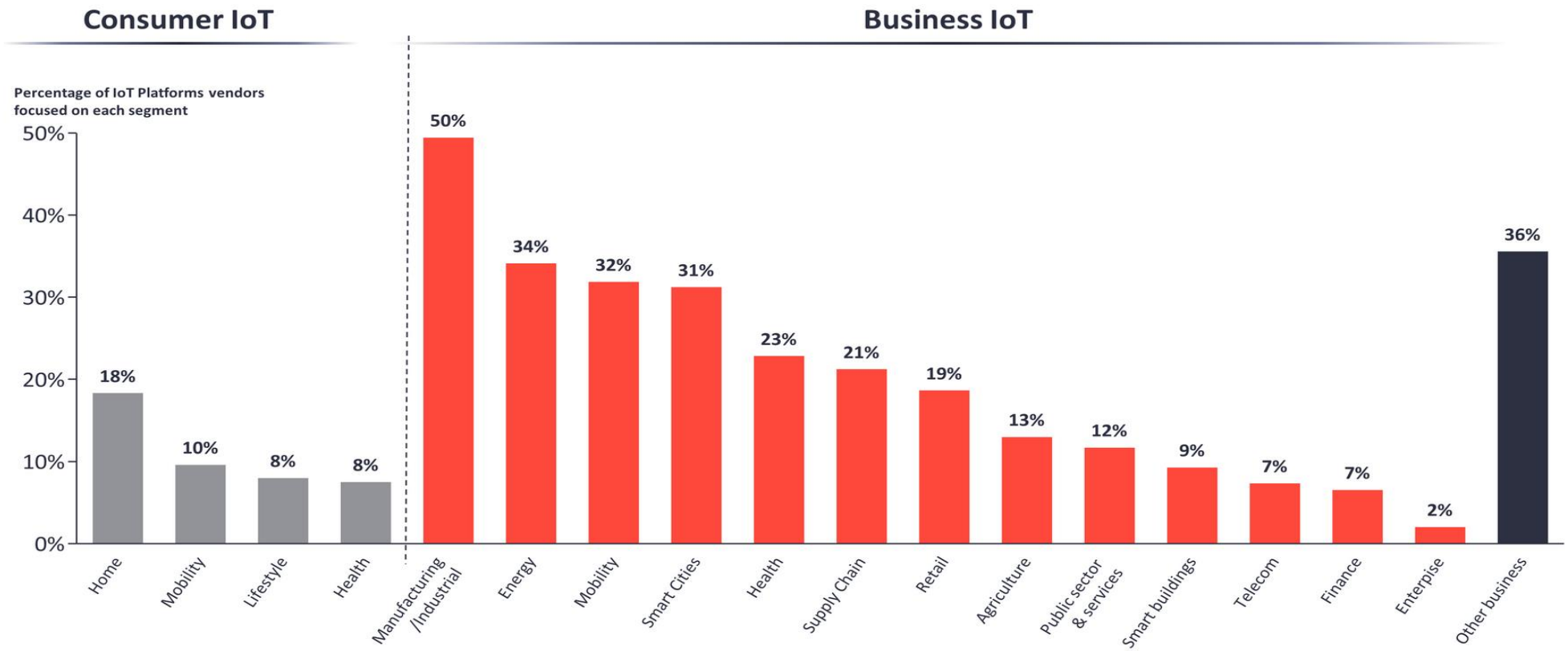
40+ example providers

BOSCH, amazon, HUAWEI, Deutsche Telekom, SIEMENS, GE, IoT, Hitachi Vantara, ORACLE INTERNET OF THINGS CLOUD SERVICE, software AG Freedom as a Service, orange, ptc, Device Insight, swisscom, FUJITSU, EVRYTHNG, Particle, KAA, UPTAKE, myDevices, cisco Cisco Kinetic, ADAMOS, accenture, Microsoft, Sprint, SAP, IBM, actility, ORACLE, vodafone, FOGHORN, verizon, AT&T, Google, Alibaba Group 阿里巴巴集团, Telefónica, EUROTECH Imagine. Build. Succeed., Schneider Electric, ERICSSON, Telit, sparkcognition, SIERRA WIRELESS

Source(s): IoT Analytics Research

AKRAINO

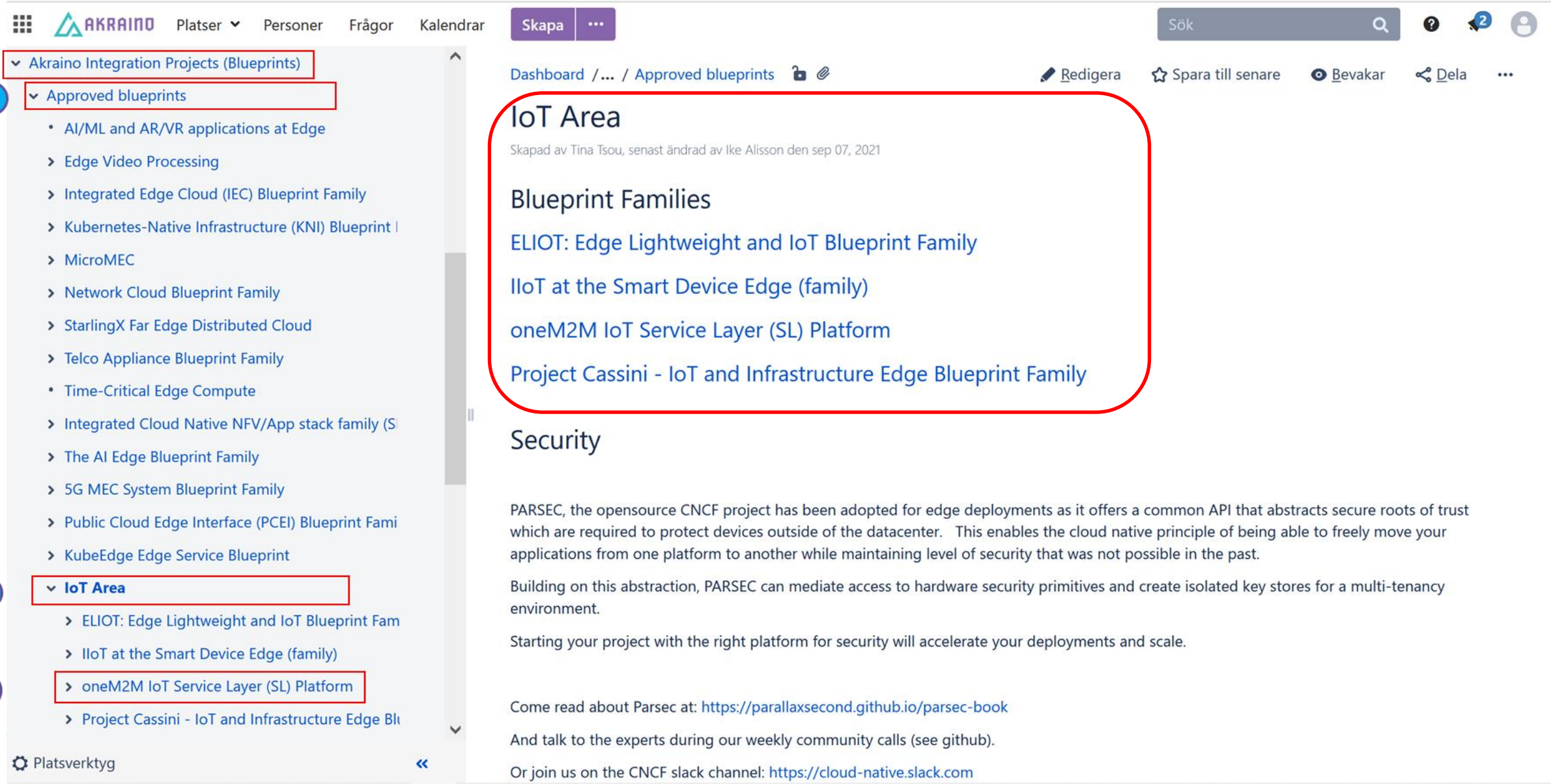
Number of Identified IoT Platforms – By industry (Dec 2019)



Note: Percentages do not add up to 100% as most companies focus on several segments **Source:** IoT Analytics 2020 List of IoT Platform Companies, n=620

1.1.5.1 Akraino IoT Area - 1

https://wiki.akraino.org/display/AK/IoT+Area



ELIOT – Overview

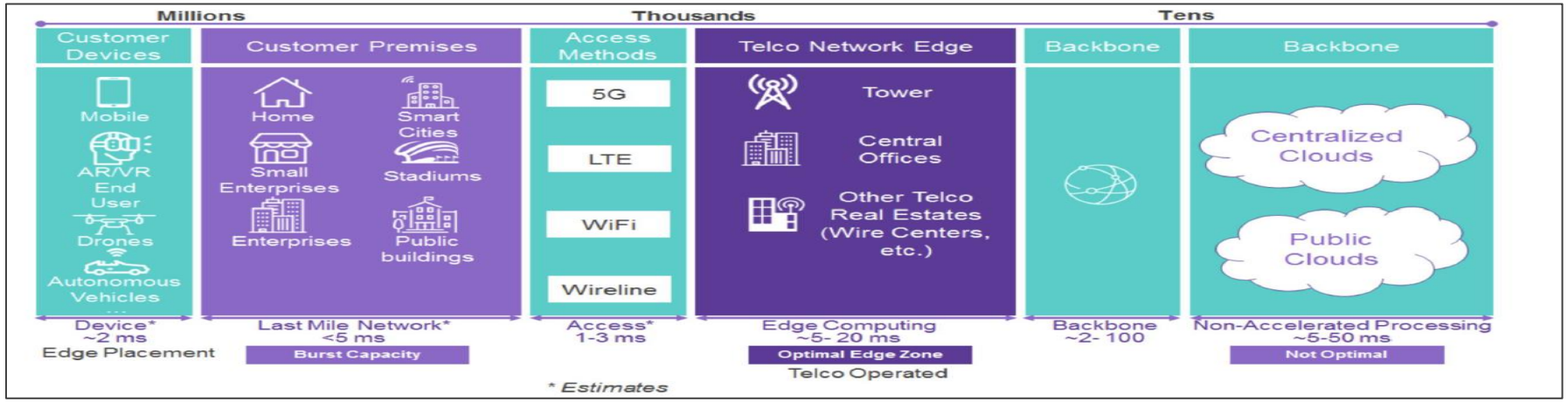
ELIOT is project under Akraino approved blueprint family.It intends to develop a fully integrated edge network infrastructure and running edge computing applications on lightweight Edge Nodes.

ELIOT targets on making the edge node a lightweight software stack which can be deployed on resource constraint edge devices like IOT-Gateway and uCPE, by leveraging lightweight OS, container running environment and container orchestration applications.

ELIOT BP family target 2 use case:

- IoT gateway
- SD-WAN, WAN edge, uCPE

Edges



ELIOT target edge

Akraino Blueprint: Smart Cities

The purpose of Smart Cities blueprint is to provide edge computing platform base on Arm Soc, Improve deployment flexibility and security in the edge computing. The high-level relationship between the functional domains is shown in the figure below:

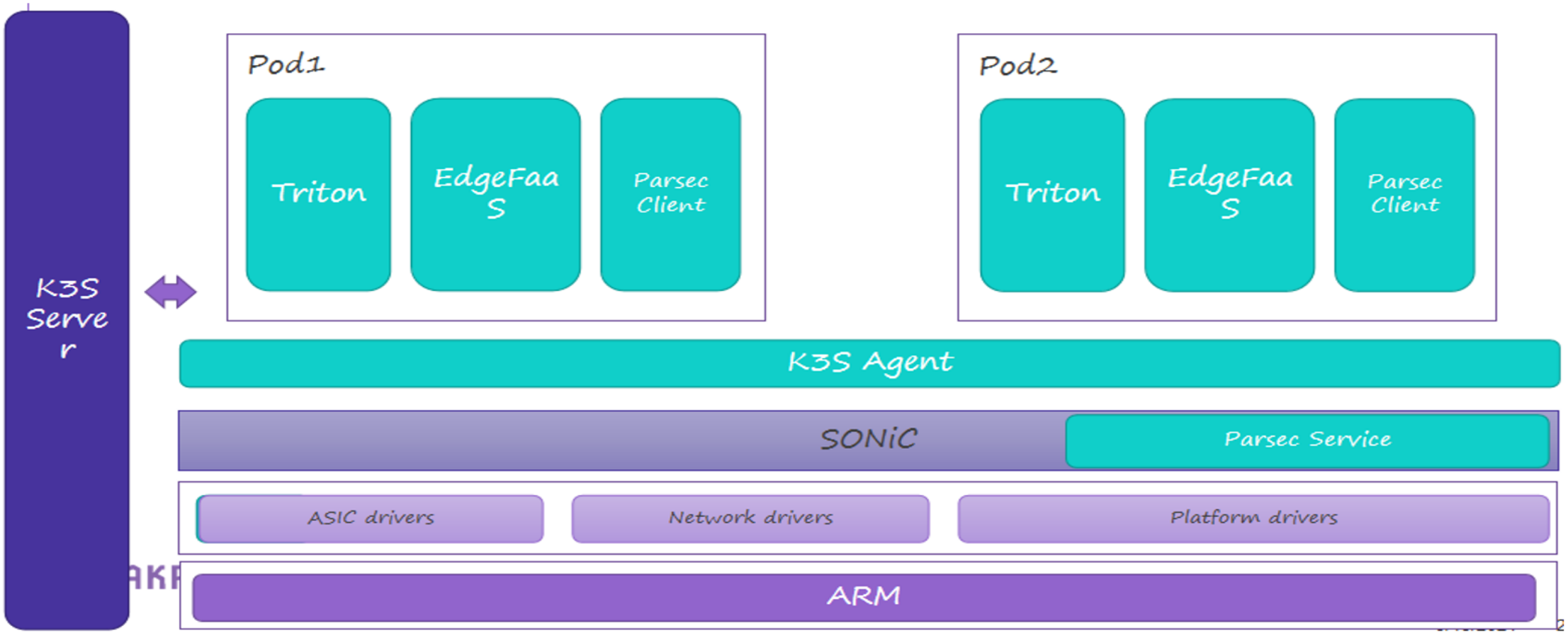
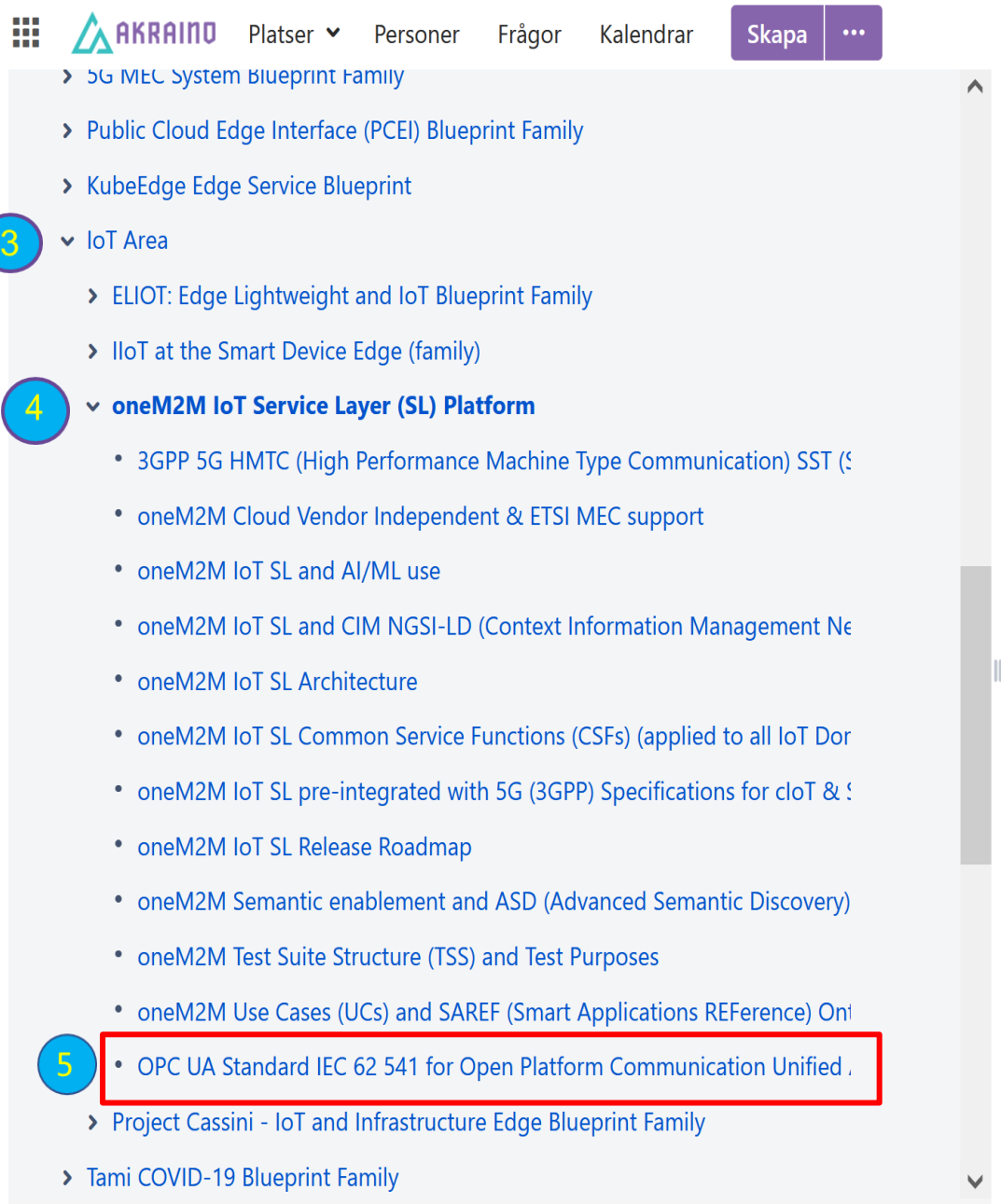


Figure 1. Smart Cities Functional Domains.

1.1.5.1 Akraino IoT Area - 2



oneM2M IoT Service Layer (SL) Platform

Skapad av Ike Alisson, senast ändrad den sep 04, 2021

oneM2M Overview

The oneM2M Global Organization creates Technical Specifications (TSs) to ensure that Machine-to-Machine (M2M) Communications can effectively operate on a Worldwide scale.

Seven (7) of the World's leading Information and Communications Technology (ICT) Standards Development Organizations (SDOs) launched in July 2012 a new Global Organization to ensure the most efficient Deployment of Machine-to-Machine (M2M) Communications Systems.

The new organization, called **oneM2M**, develops specifications to ensure the Global Functionality of M2M—allowing a range of Industries to effectively take advantage of the benefits of this emerging Technology.

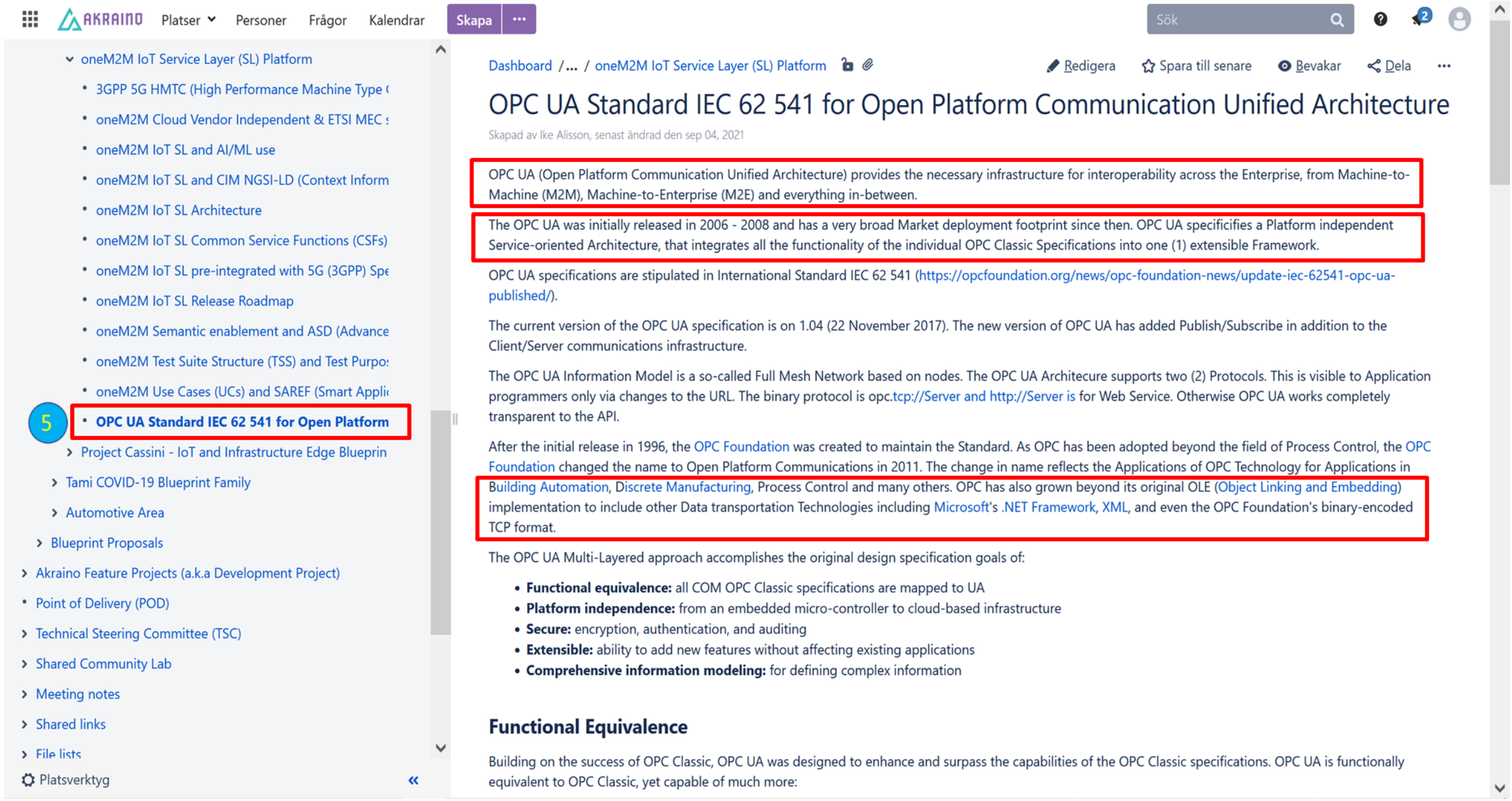
The seven (7) majors ICT SDO founders of oneM2M are:

- The European Telecommunications Standards Institute (ETSI) , Europe
- The Association of Radio Industries and Businesses (ARIB), Japan
- The Telecommunication Technology Committee (TTC), Japan
- The Alliance for Telecommunications Industry Solutions (ATIS), USA
- The Telecommunications Industry Association (TIA), USA
- The China Communications Standards Association (CCSA), China
- The Telecommunications Technology Association (TTA), Korea

The members of the organization are devoted to developing Technical Specifications and Reports to ensure M2M Devices can successfully communicate on a Global scale.

The oneM2M Standardization work is split in five (5) WG:

1.1.5.1 Akraino IoT Area - 3



1.1.5.1 Akraino IoT Area - 4

AKRAINO Platser Personer Frågor Kalendrar Skapa ••• Sök 2

- oneM2M IoT Service Layer (SL) Platform
  - 3GPP 5G HMTC (High Performance M
  - oneM2M Cloud Vendor Independent
  - oneM2M IoT SL and AI/ML use
  - oneM2M IoT SL and CIM NGSI-LD (C
  - oneM2M IoT SL Architecture
  - oneM2M IoT SL Common Service Fur
  - oneM2M IoT SL pre-integrated with S
  - oneM2M IoT SL Release Roadmap
  - oneM2M Semantic enablement and A
  - oneM2M Test Suite Structure (TSS) ar
  - oneM2M Use Cases (UCs) and SAREF
- 5 **OPC UA Standard IEC 62 541 for O**
- Project Cassini - IoT and Infrastructure I
- Tami COVID-19 Blueprint Family

AKRAINO

For further information on the OPC UA, please see attached below the OPC UA Open IEC 62 541 (current) Documentation from Jan 2021.

Title	Description	Title	Description
Part 1: Overview and Concepts	This specification provides a high-level introduction to the Unified Architecture technology covering: More >	Part 11: Historical Access	This specifications describes how data can be archived and retrieved from a Historian/database, covering: More >
Part 2: Security Model	This specification describes OPC UA Security, covering: Introduction of the security objectives and More >	Part 12: Discovery and Global Services	This specification describes how UA products can be discovered and managed on a computer, network infrastructure, or More >
Part 3: Address Space Model	This specification provides a detailed description of an address space within an OPC UA Server, for OPC UA Clients to More >	Part 13: Aggregates	This specification describes the use of Aggregate functions for UA applications, covering: The concepts of More >
Part 4: Services	This specification is the most important of all OPC UA specifications, covering: The UA Services More >	Part 14: PubSub	This specification defines the OPC Unified Architecture (OPC UA) PubSub communication model. The PubSub communication More >
Part 5: Information Model	This specification provides a detailed description of how the OPC UA address space, nodes, and references are used to More >	Part 15: Safety	The specification "OPC UA Safety" describes services and protocols for the exchange of data using OPC UA mechanisms. More >
Part 6: Mappings	This specification describes how data and information are transferred between OPC UA Servers and Clients, covering: More >	Part 17: Alias Names	This specification provides a definition of AliasNames functionality. AliasNames provide a manner of configuring and More >
Part 7: Profiles	This specification describes categories of behaviors that OPC UA Servers and Clients can implement, covering: More >	Part 19: Dictionary Reference	This specification defines an Information Model of the OPC Unified Architecture. The Information Model describes the More >
Part 8: Data Access	This specification describes Data Access applications, covering: Overview and concepts of Data Access and More >	Part 100: Device Information Model	Companion Specification featuring an Information Model for Devices. The information model specification More >
Part 9: Alarms and Conditions	This specification describes the Alarms & Conditions applications, covering: Overview and concepts of More >	Part 200: Industrial Automation Model	This specification contains modelling concepts used in industrial automation. Version 1.00 contains modelling concepts More >
Part 10: Programs	This specification describes Programs and how they can be used in OPC UA applications, covering: Concepts of More >	Errata and Amendments	These documents contain changes to the OPC UA Specifications. These changes may impact interoperability and compliance. More >
		Specification Release Candidates for Review	These documents are draft and release candidate versions of OPC UA Specifications for member review. Information on how More >
		OPC UA Companion Specification Template	The companion specification template (OPC 11020) together with guidelines (OPC 11021) can be found here.

OPC FOUNDATION
The Industrial Interoperability Standard™

open62541

open62541 Documentation

Release 1.2.0-rc2-44-ge5eba7bd

The open62541 authors

January 06, 2021

1.1.5.1 Akraino IoT Area - 5 (OPC UA IEC 62 541)

2.1 Building the Library

2.1.1 Building with CMake on Ubuntu or Debian

```
sudo apt-get install git build-essential gcc pkg-config cmake python

# enable additional features
sudo apt-get install cmake-curses-gui # for the ccmake graphical interface
sudo apt-get install libmbedtls-dev # for encryption support
sudo apt-get install check libsubunit-dev # for unit tests
sudo apt-get install python-sphinx graphviz # for documentation generation
sudo apt-get install python-sphinx-rtd-theme # documentation style

cd open62541
mkdir build
cd build
cmake ..
make

# select additional features
ccmake ..
make

# build documentation
make doc # html documentation
make doc_pdf # pdf documentation (requires LaTeX)
```

2.1.2 Building with CMake on Windows

Here we explain the build process for Visual Studio (2013 or newer). To build with MinGW, just replace the compiler selection in the call to CMake.

- Download and install
  - Python 2.7.x (Python 3.x works as well): https://python.org/downloads
  - CMake: http://www.cmake.org/cmake/resources/software.html
  - Microsoft Visual Studio: https://www.visualstudio.com/products/visual-studio-community-vs

2.3 Building the Examples

Make sure that you can build the shared library as explained in the previous steps. Even easier way to build the examples is to install open62541 in your operating system (see *Installing open62541*).

Then the compiler should automatically find the includes and the shared library.

```
cp /path-to/examples/tutorial_server_firststeps.c . # copy the example server
gcc -std=c99 -o server tutorial_server_firststeps.c -lopen62541
```

2.4 Building for specific architectures

The open62541 library can be build for many operating systems and embedded systems. This document shows a small excerpt of already tested architectures. Since the stack is only using the C99 standard, there are many more supported architectures.

A full list of implemented architecture support can be found in the arch folder.

2.4.1 Windows, Linux, MacOS

These architectures are supported by default and are automatically chosen by CMake.

Have a look into the previous sections on how to do that.

2.4.2 freeRTOS + LwIP

Credits to @cabralfortiss

This documentation is based on the discussion of the PR https://github.com/open62541/open62541/pull/2511. If you have any doubts, please first check the discussion there.

3.2 Prebuilt packages

3.2.1 Pack branches

Github allows you to download a specific branch as .zip package. Just using this .zip package for open62541 will likely fail:

- CMake uses `git describe --tags` to automatically detect the version string. The .zip package does not include any git information
- Specific options during the build stack require additional git submodules which are not inlined in the .zip

Therefore we provide packaging branches. They have the prefix *pack/* and are automatically updated to match the referenced branch.

Here are some examples:

- pack/master.zip
- pack/1.0.zip

These pack branches have inlined submodules and the version string is hardcoded. If you need to build from source but do not want to use git, use these specific pack versions.

3.2.2 Prebuilt binaries

You can always find prebuilt binaries for every release on our Github Release Page.

Nightly single file releases for Linux and Windows of the last 50 commits can be found here: https://open62541.org/releases/

3.2.3 Debian

Debian packages can be found in our official PPA:

- Daily Builds (based on master branch): https://launchpad.net/~open62541-team/+archive/ubuntu/daily
- Release Builds (starting with Version 0.4): https://launchpad.net/~open62541-team/+archive/ubuntu/ppa

Install them with:

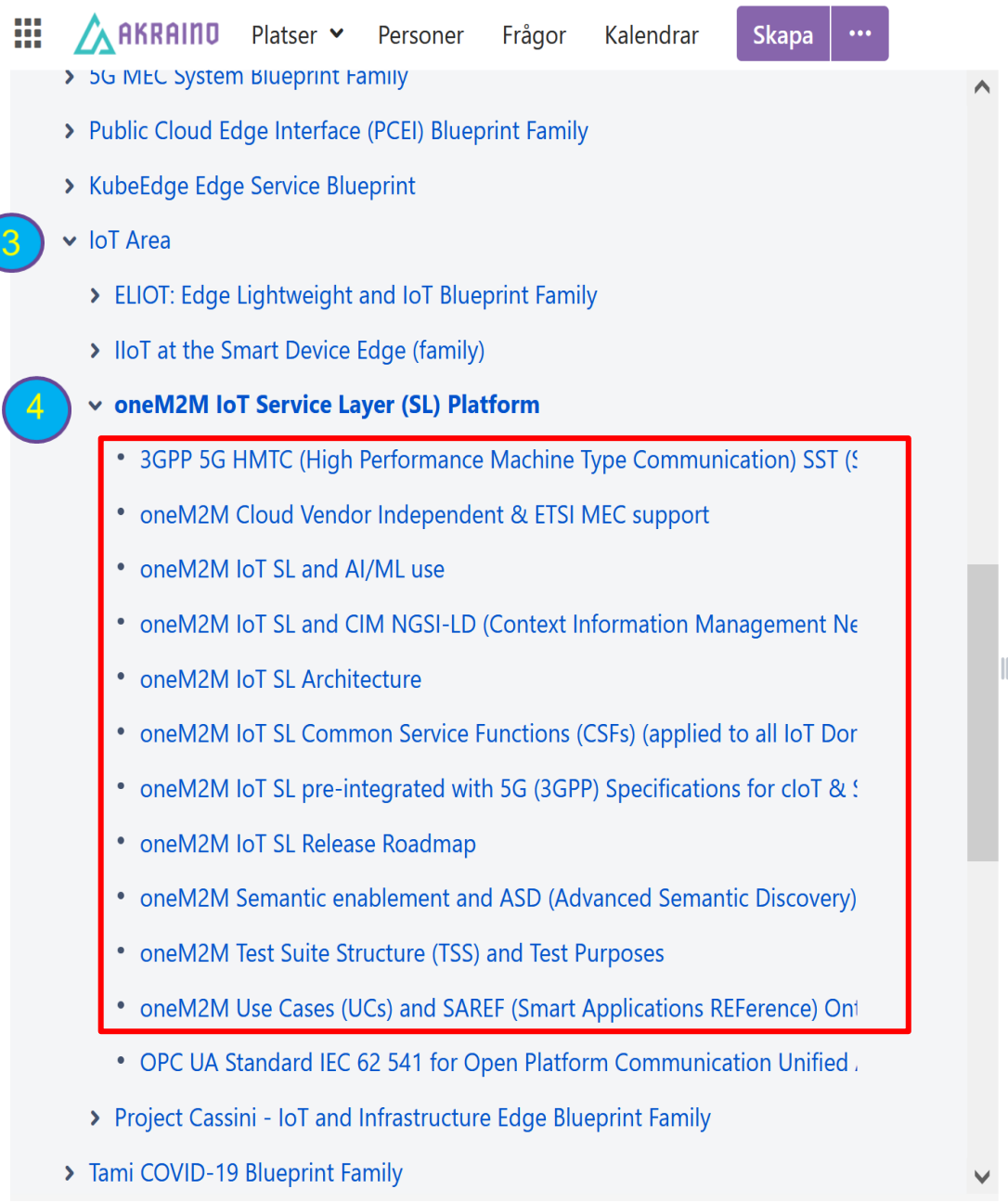
```
sudo add-apt-repository ppa:open62541-team/ppa
sudo apt-get update
sudo apt-get install libopen62541-1-dev
```

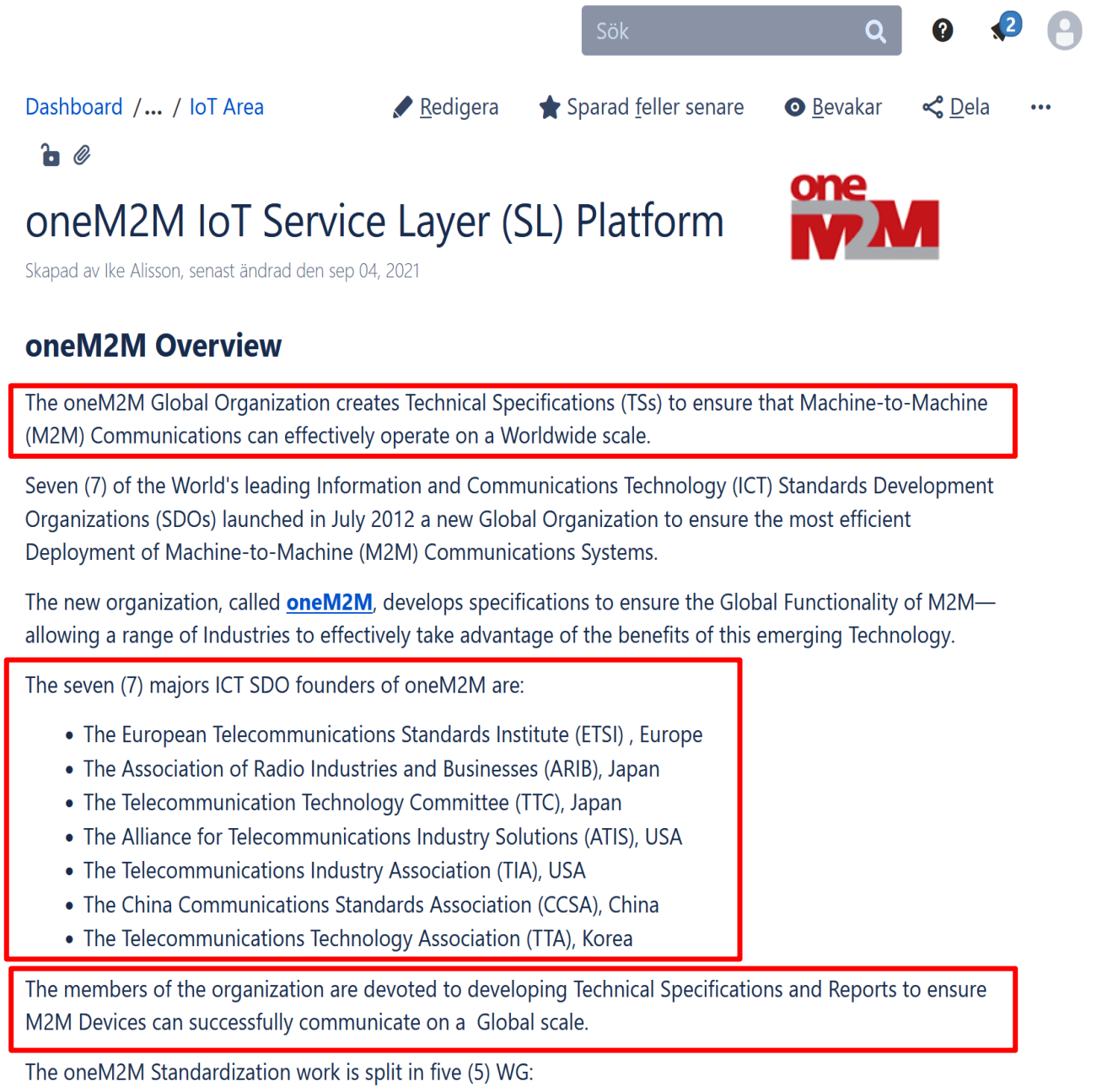
3.2.4 Arch

Arch packages are available in the AUR:

- Stable Builds: https://aur.archlinux.org/packages/open62541/
- Unstable Builds (current master): https://aur.archlinux.org/packages/open62541-git/
- In order to add custom build options (*Build Options*), you can set the environment variable `OPEN62541_CMAKE_FLAGS`

1.1.5.1 Akraino IoT Area - 6





1.1.5.1 Akraino IoT Area - 7

- IoT Area
  - ELIOT: Edge Lightweight and IoT Blueprint Fam
  - IIoT at the Smart Device Edge (family)
  - **oneM2M IoT Service Layer (SL) Platform**
    - 3GPP 5G HMTC (High Performance Machine
    - oneM2M Cloud Vendor Independent & ETSI
    - oneM2M IoT SL and AI/ML use
    - oneM2M IoT SL and CIM NGSI-LD (Context I
    - oneM2M IoT SL Architecture
    - oneM2M IoT SL Common Service Functions
    - oneM2M IoT SL pre-integrated with 5G (3GF
    - oneM2M IoT SL Release Roadmap
    - oneM2M Semantic enablement and ASD (Ac
    - oneM2M Test Suite Structure (TSS) and Test I
    - oneM2M Use Cases (UCs) and SAREF (Smart
    - OPC UA Standard IEC 62 541 for Open Platfc
  - Project Cassini - IoT and Infrastructure Edge Bl
  - Tami COVID-19 Blueprint Family
  - Automotive Area
- Blueprint Proposals
- Akraino Feature Projects (a.k.a Development Project)

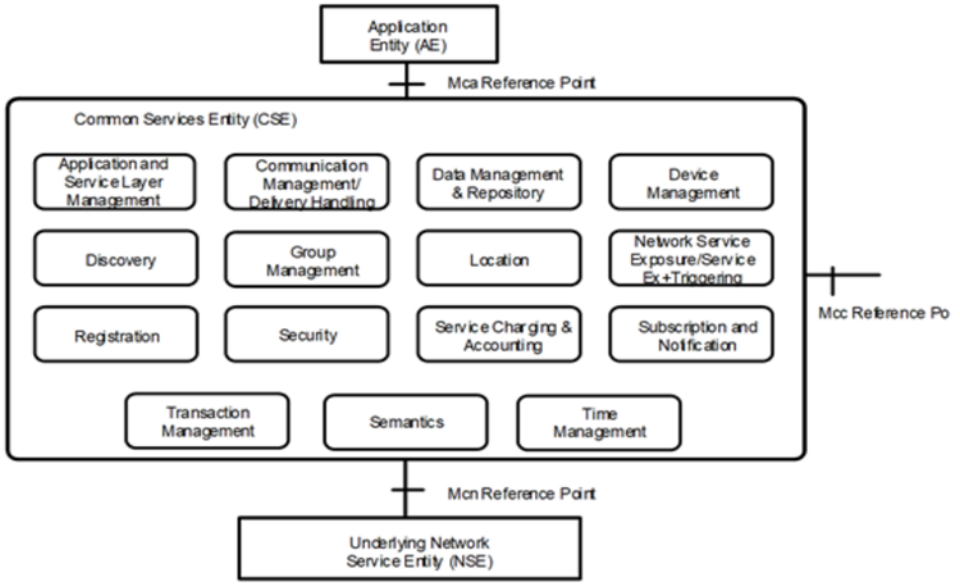
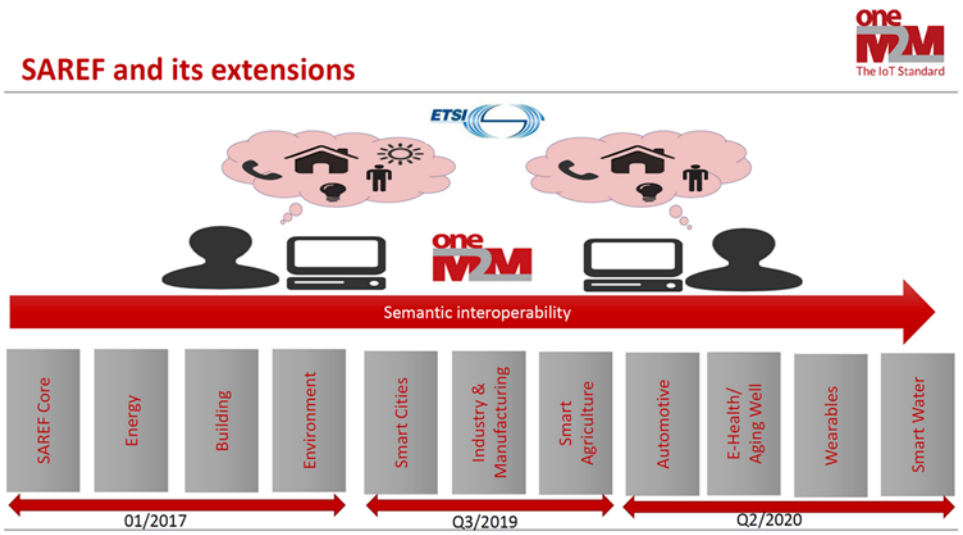


Fig. 6.2.0-1: Common Service Functions



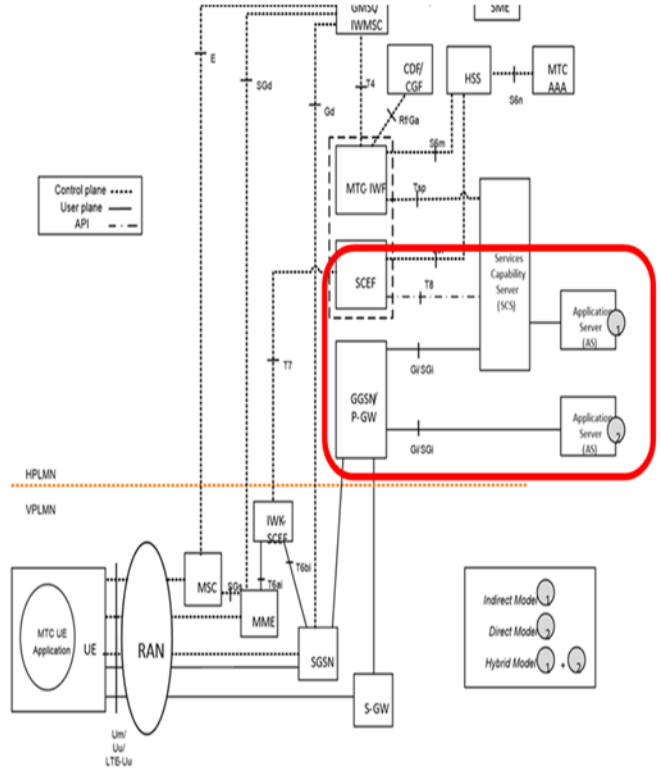
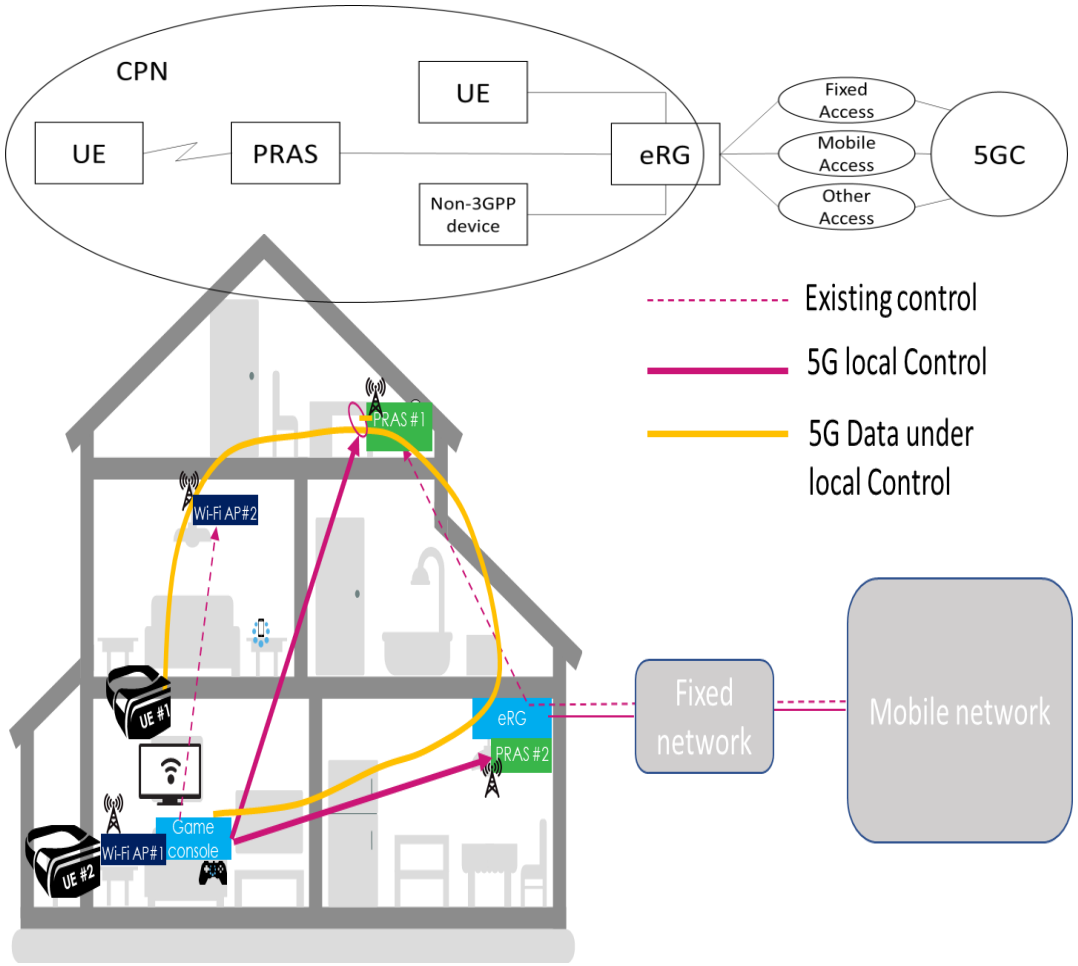


Figure 4.2-1b: 3GPP Architecture for Machine-Type Communication (Roaming)

1.1.5.1 Akraino IoT Area - 8

Personal IoT Networks (PINs)



AKRAINO

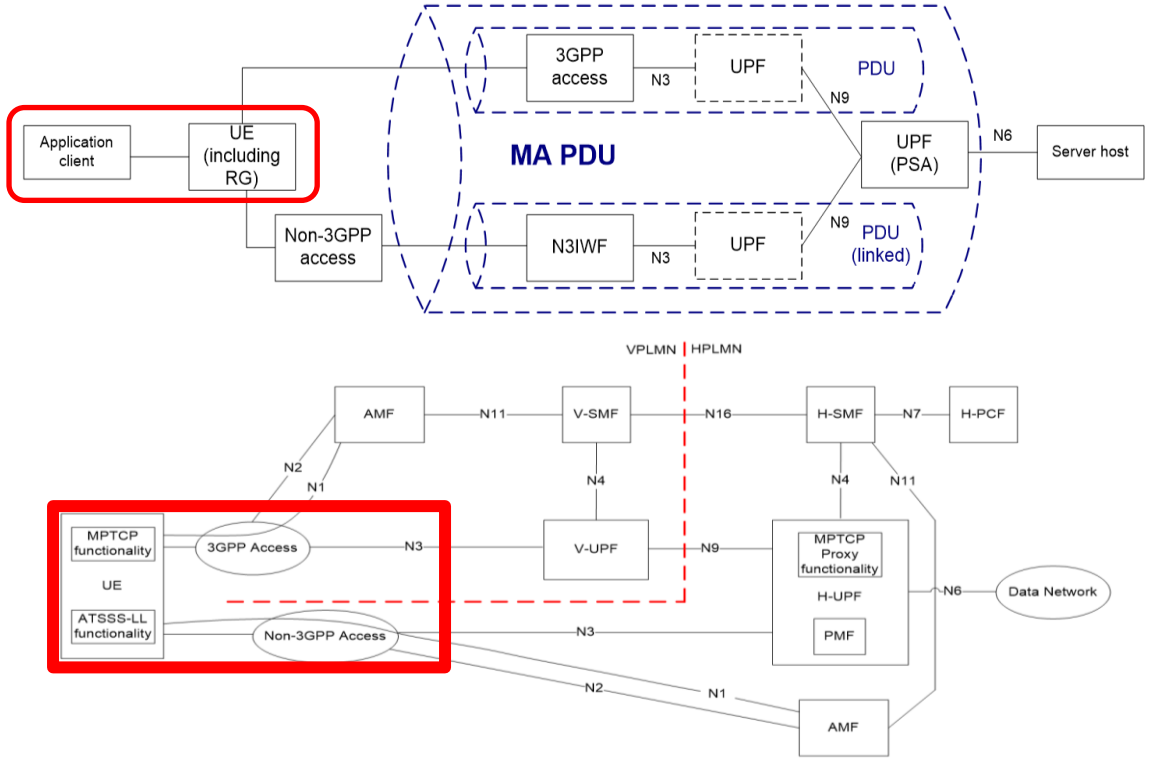


Figure 4.2.10-3: Roaming with Home-routed architecture for ATSSS support (UE registered to different PLMNs)

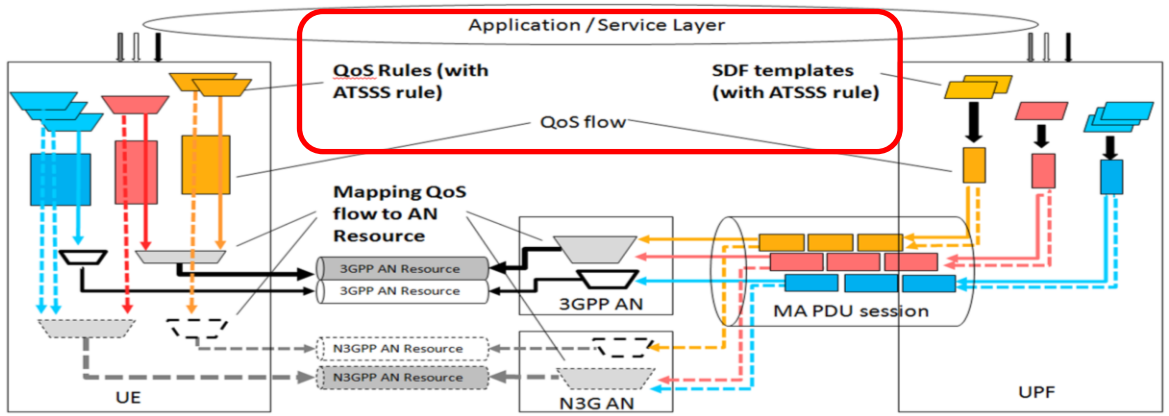


Figure 6.3.1.2-1: The traffic splitting based on the QoS rule (with ATSSS rule)

3GPP RAN Rel-16 progress and Rel-17 potential work areas

July 18, 2019

https://www.3gpp.org/news-events/2058-ran-rel-16-progress-and-rel-17-potential-work-areas

Slide 7

Release 16 progressing towards completion

- 5G V2X
  - Targeting advanced use cases beyond LTE V2X
- Industrial IoT and URLLC enhancements
  - Adding 5G NR capabilities for full wired Ethernet replacement in factories: Time Sensitive networking, etc... with high reliability
- 5G NR operation in unlicensed bands
  - Includes both Licensed Assisted Access (LAA), as well as Standalone Unlicensed operation
- System improvements and enhancements
  - Positioning
  - MIMO enhancements
  - Power Consumption improvements

 7

1.1.5.1 Akraino IoT Area - 10

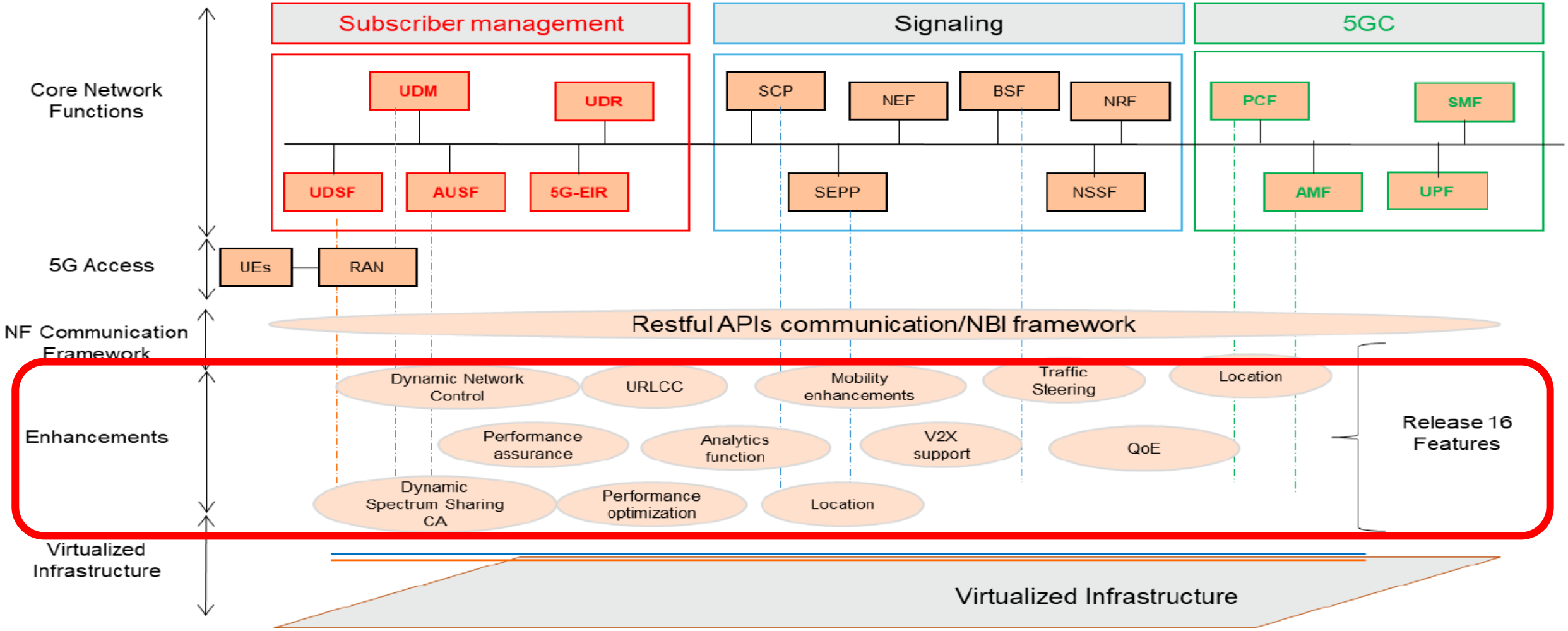


Figure 2-6: Release 16 5G features and enhancements supporting verticals

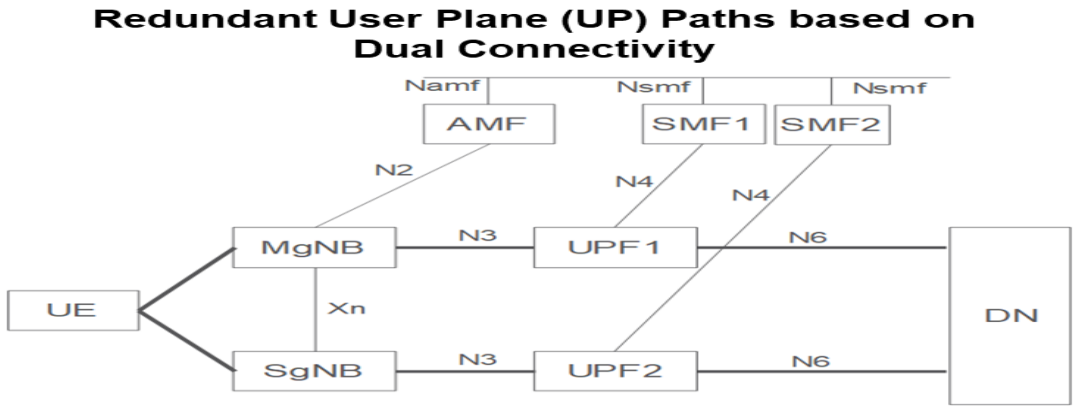


Figure 6.1.1-2: Solution architecture

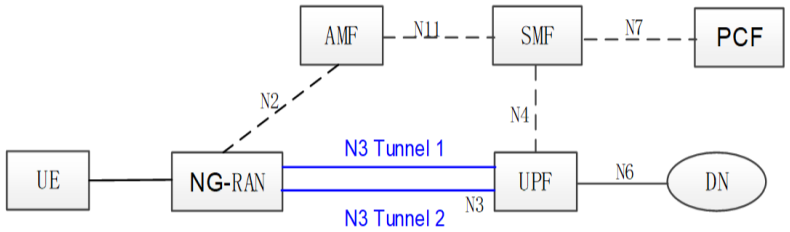


Figure 6.4.1-1: Redundant transmission with two N3 tunnels between the UPF and a single NG-RAN node

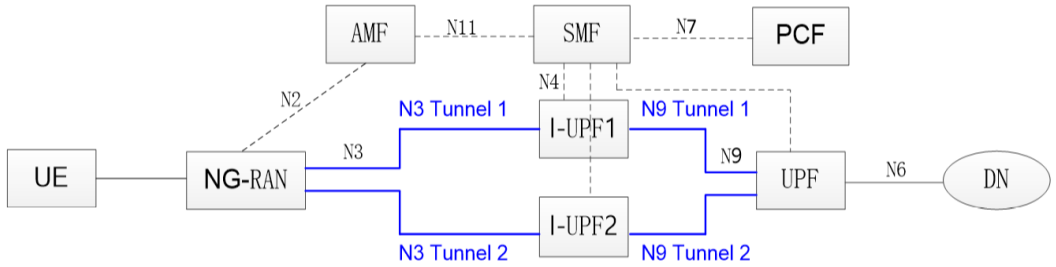


Figure 6.4.1-2 Two N3 and N9 tunnels between NG-RAN and UPF for redundant transmission

Static approach:

This applies to both IP and Ethernet PDU sessions. The solution is illustrated in the Figure below:

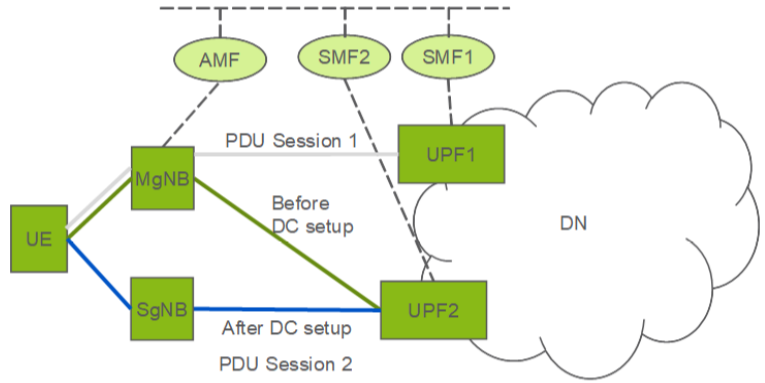


Figure 6.1.1-3: Static UPF selection

Dynamic approach:

This applies to Ethernet PDU Sessions. The solution is illustrated in the Figure below:

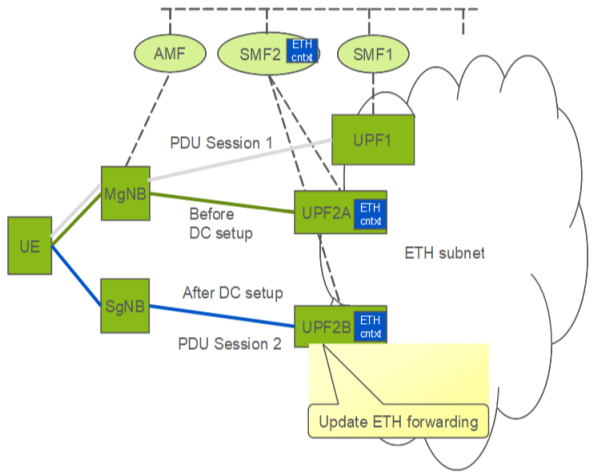
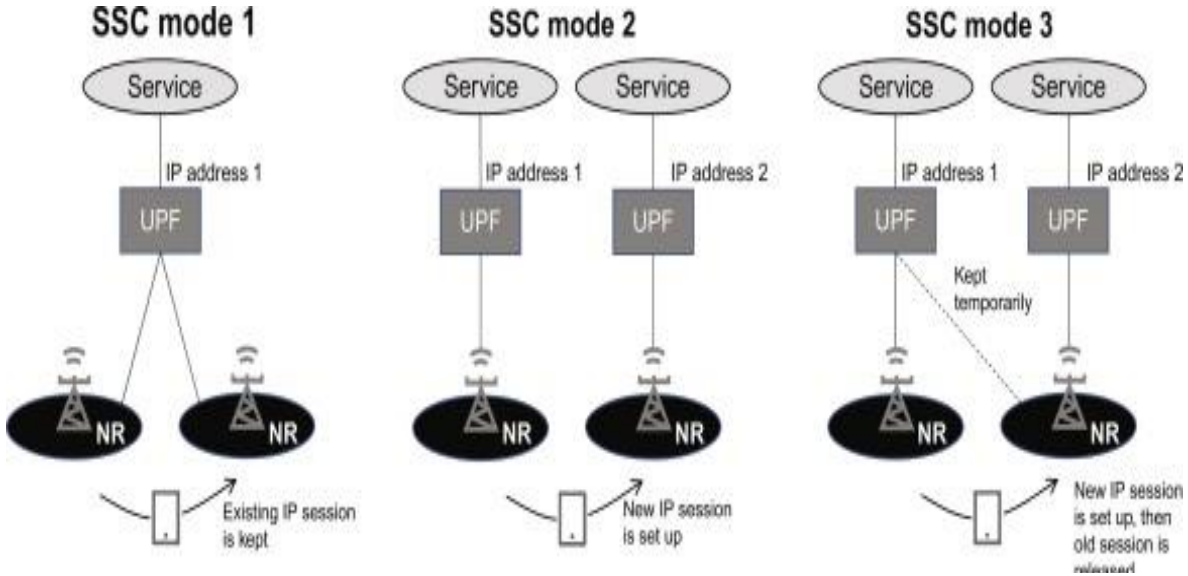


Figure 6.1.1-4: Dynamic UPF Selection: anchor change after DC setup for Ethernet PDU Sessions

1.1.5.1 Akraino IoT Area - 11



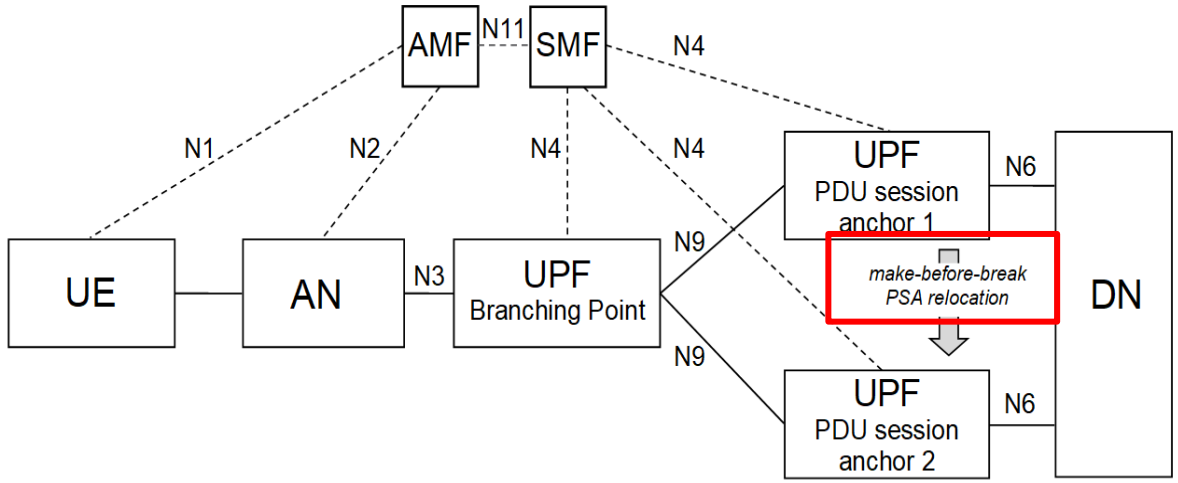


Figure 5.6.4.3-1: Multi-homed PDU Session: service continuity case

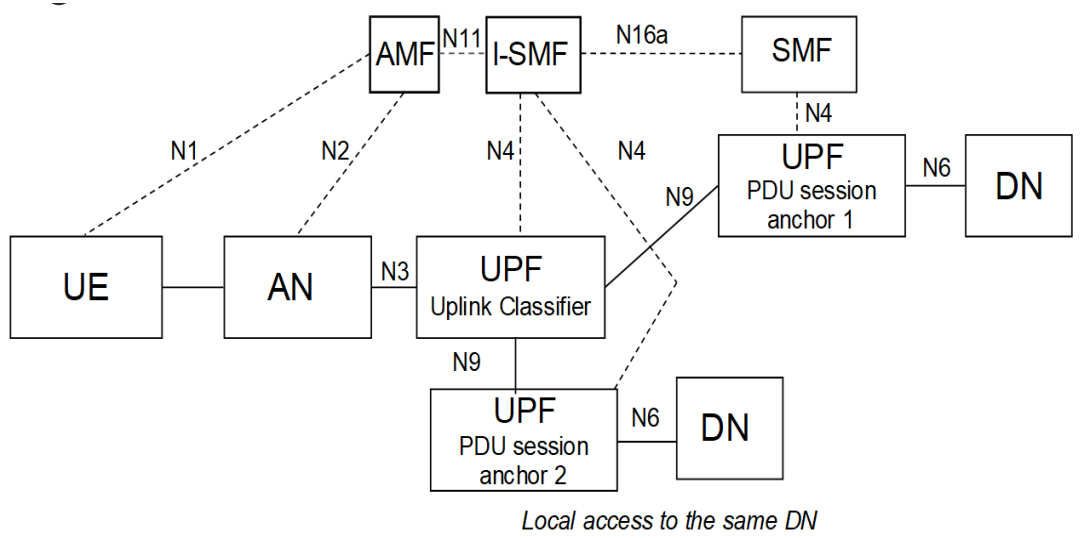


Figure 5.34.4-1: User plane Architecture for the Uplink Classifier controlled by I-SMF

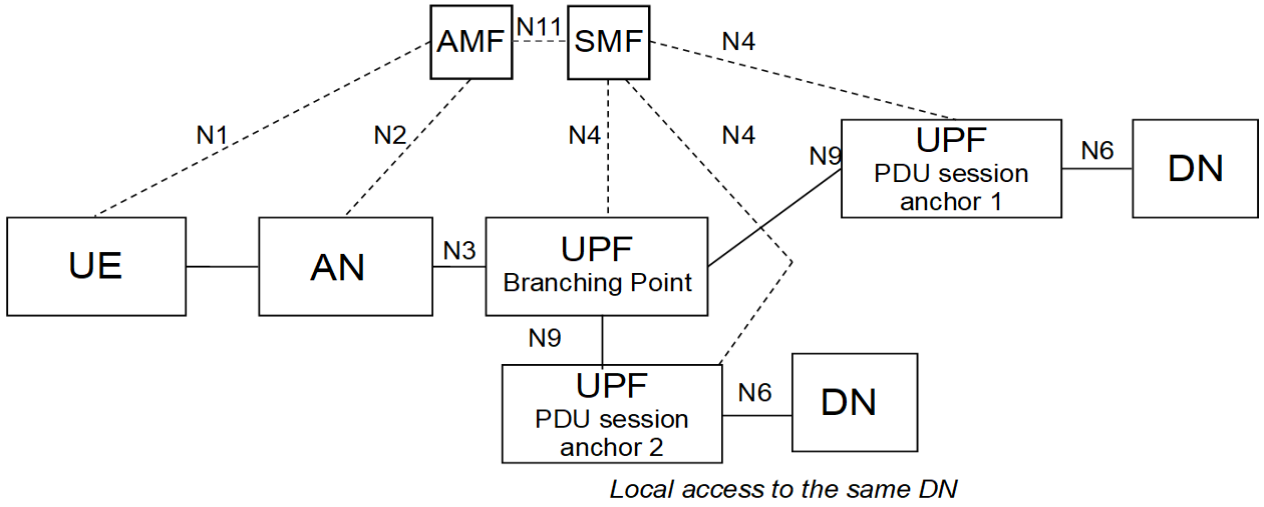


Figure 5.6.4.3-2: Multi-homed PDU Session: local access to same DN

1.1.5.1 Akraino IoT Area - 12

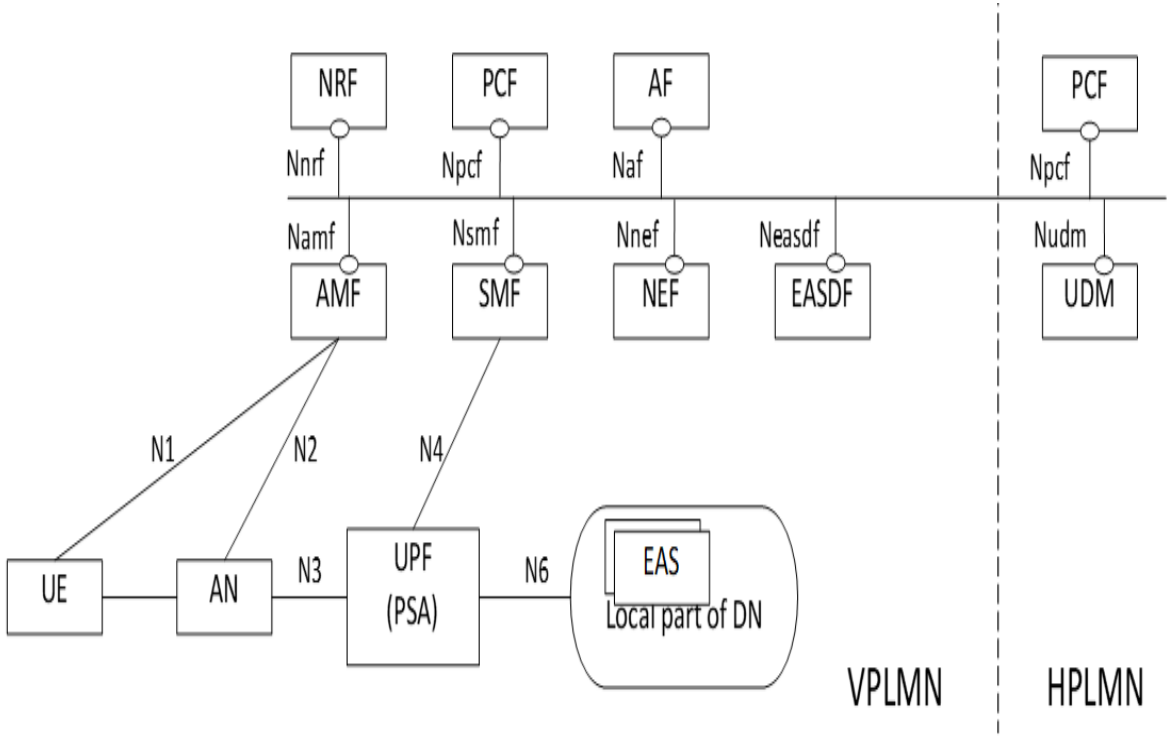


Figure 4.2-4: 5GS providing access to EAS without UL CL/BP for LBO roaming scenario

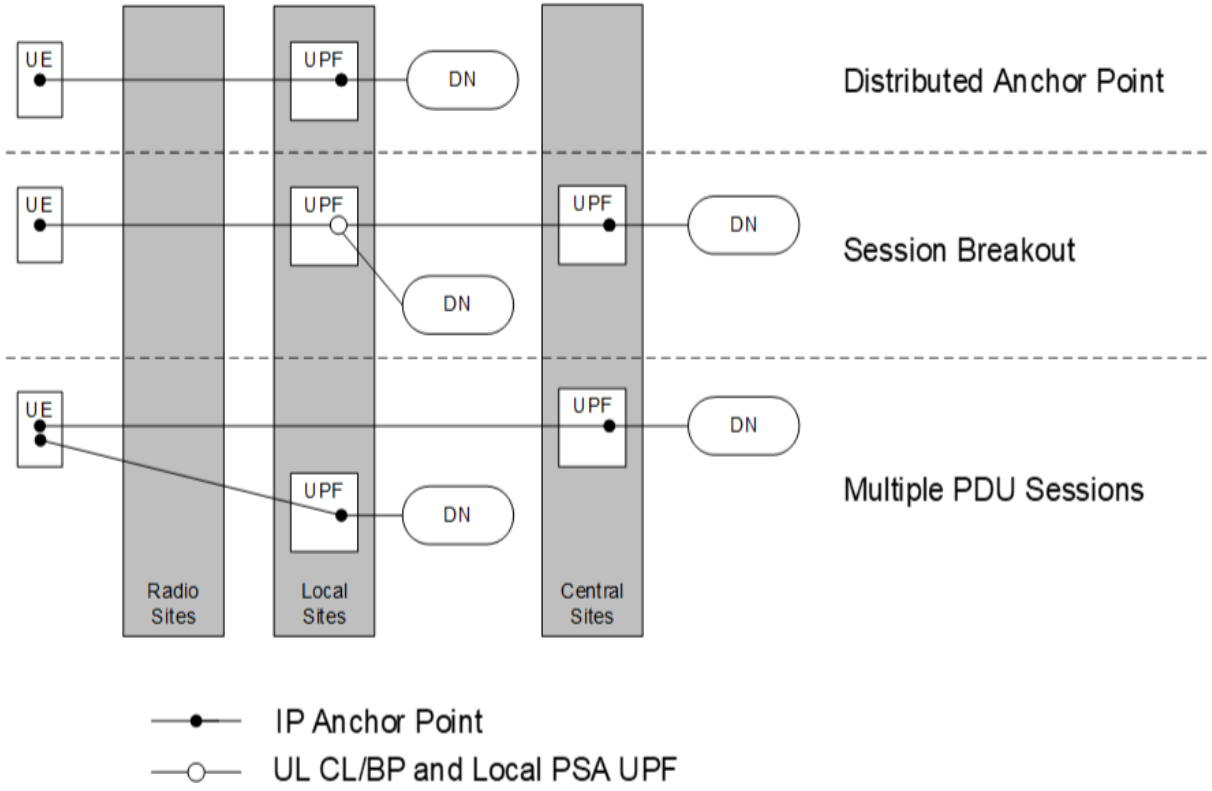


Figure 4.3-1: 5GC Connectivity Models for Edge Computing

1.1.5.1 Akraino IoT Area - 13 3GPP 5G System Idle Inactive Connected UE states

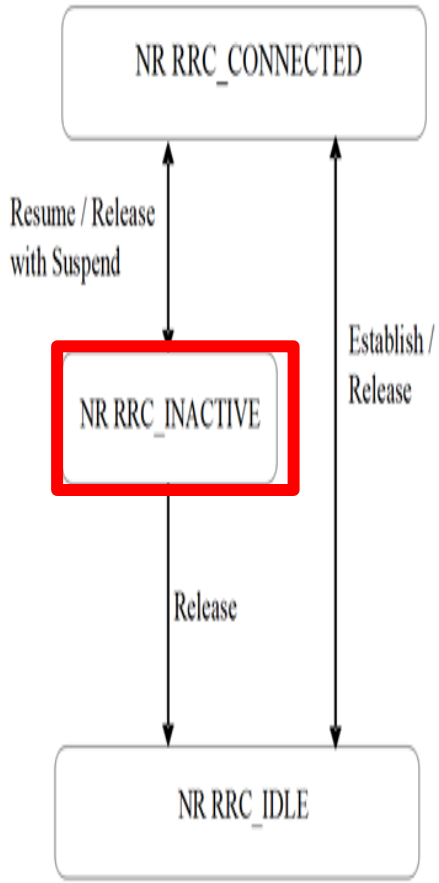


Figure 4.2.1-1: UE state machine and state transitions in NR

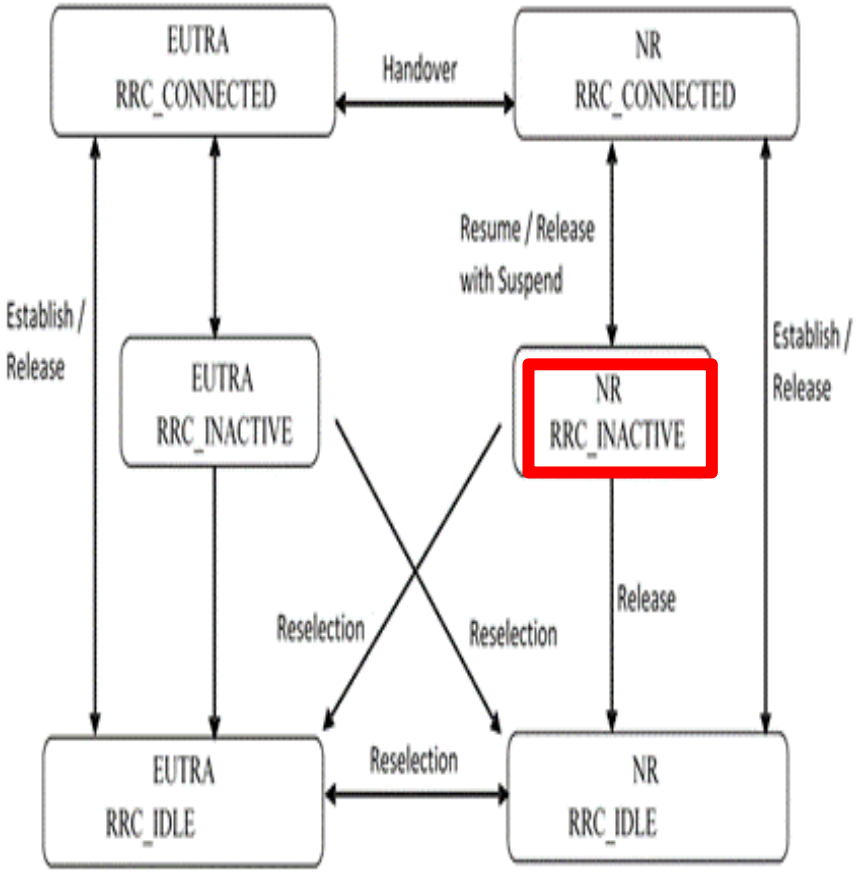


Figure 4.2.1-2: UE state machine and state transitions between NR/5GC, E-UTRA/EPC and E-UTRA/5GC

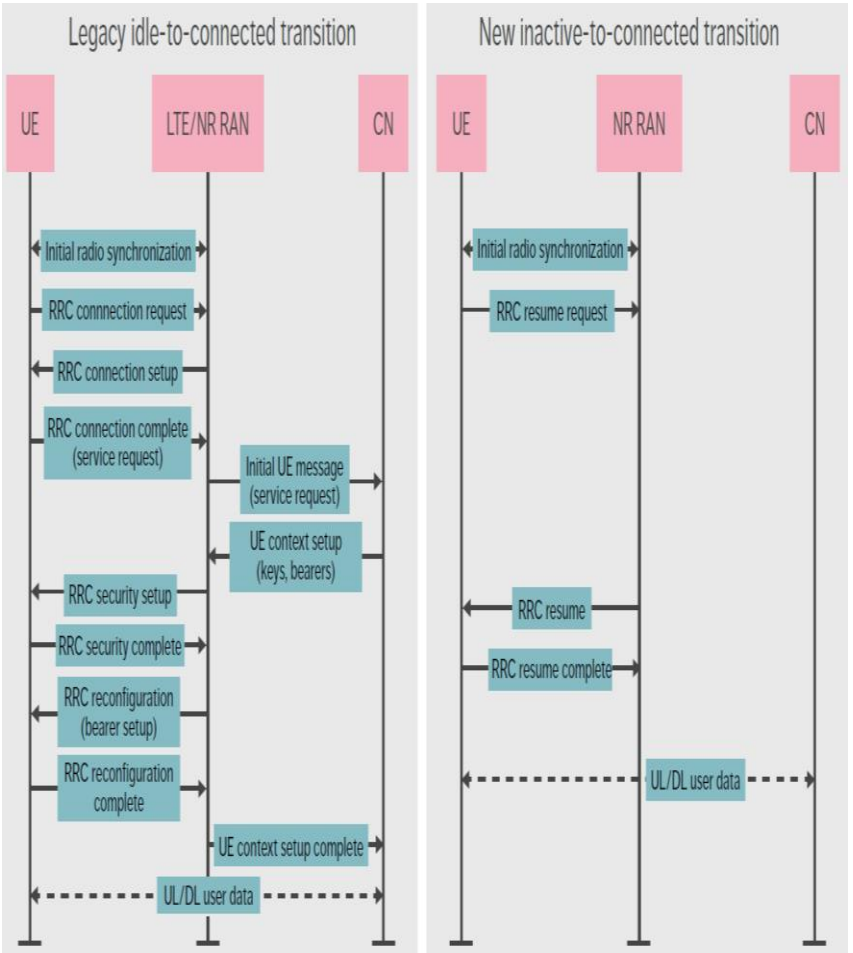


Figure 2 Comparison of Signalling involved in Legacy Idle-to-Connected transition (Left) versus Inactive-to-Connected Transition (Right)

GSMA

Operator Platform Telco Edge Requirements
Version 1.0
29 June 2021

This is a Non-binding Permanent Reference Document of the GSMA

The OPG believes that, for Operators to develop a Federated Edge Computing Platform such as the OP, Requirements must be enforceable in Contracts by a Published Set of Standards.

To this end, the OPG proposes selecting ETSI ISG MEC and 3GPP to provide a Standard Reference for an Edge Service End to End (E2E) definition.

We note that 3GPP EDGEAPP Architecture and ETSI ISG MEC Architecture could complement each other in a way that is acceptable to OPG:

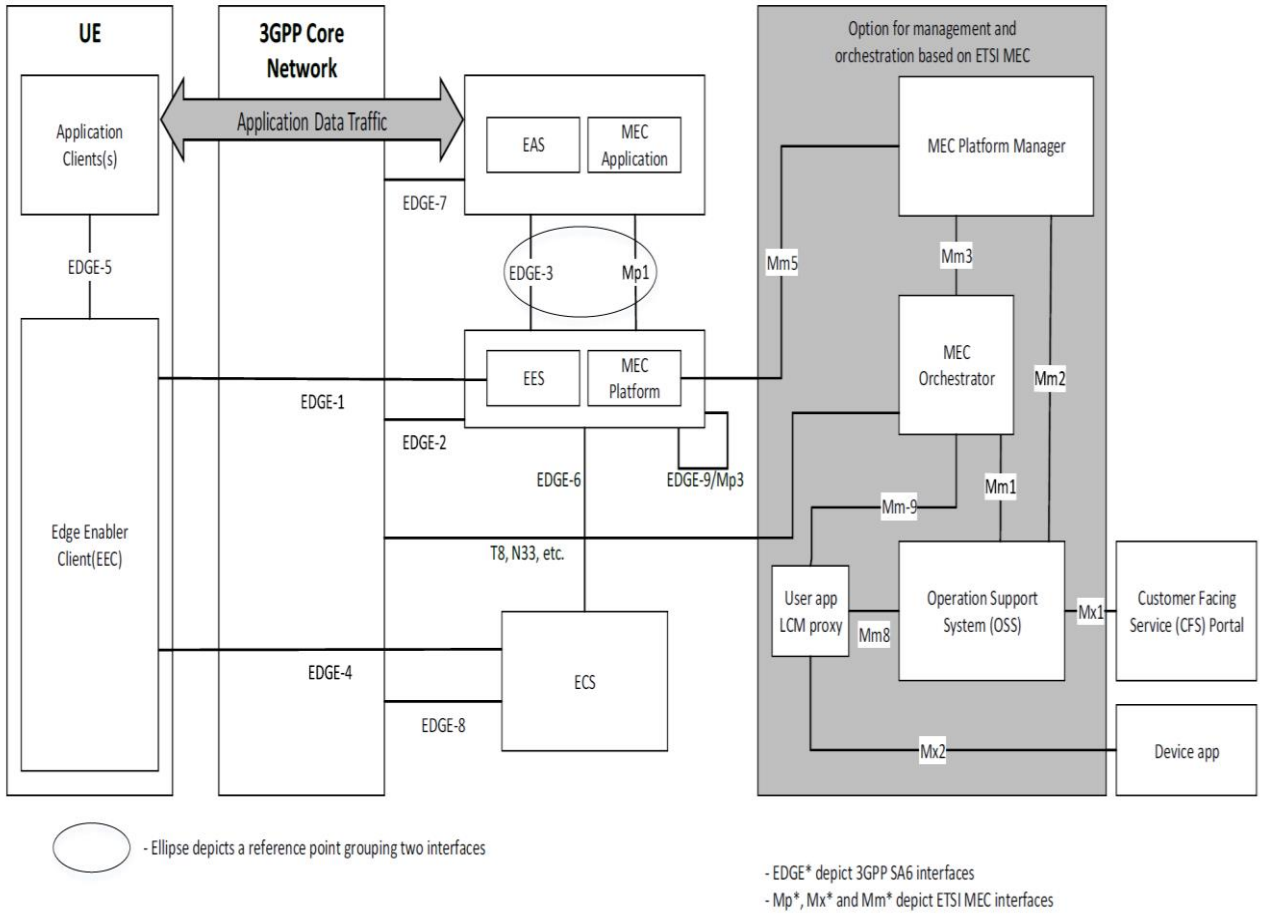
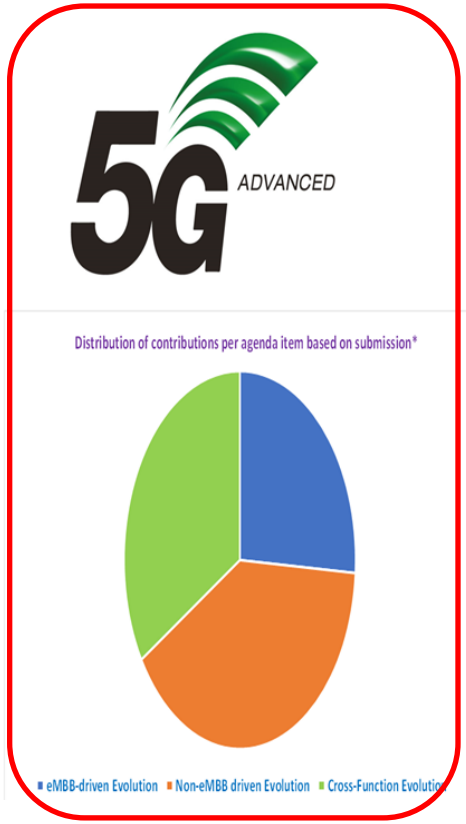


Figure 14: Relationship between ETSI ISG-MEC and 3GPP EDGEAPP architectures

AKRAINO

General

- A first discussion in 3GPP RAN on ***5G-Advanced***
  - Starting from Rel-18. Official logo is shown on the right
  - 500+ submissions from ~80 different companies/organizations
  - 1200+ checked-in participants
- Careful organization and balanced discussion of topics in three agenda items, aiming for both immediate and longer-term commercial needs
  - **Three agenda items:**
    - **eMBB (evolved Mobile BroadBand)**-driven evolution;
    - **Non-eMBB**-driven evolution;
    - **Cross-functionalities** for both eMBB-driven and non-eMBB-driven evolution
  - Some stats (very coarse!) as shown on the right → generally a **balanced evolution** in the three directions based on the submissions



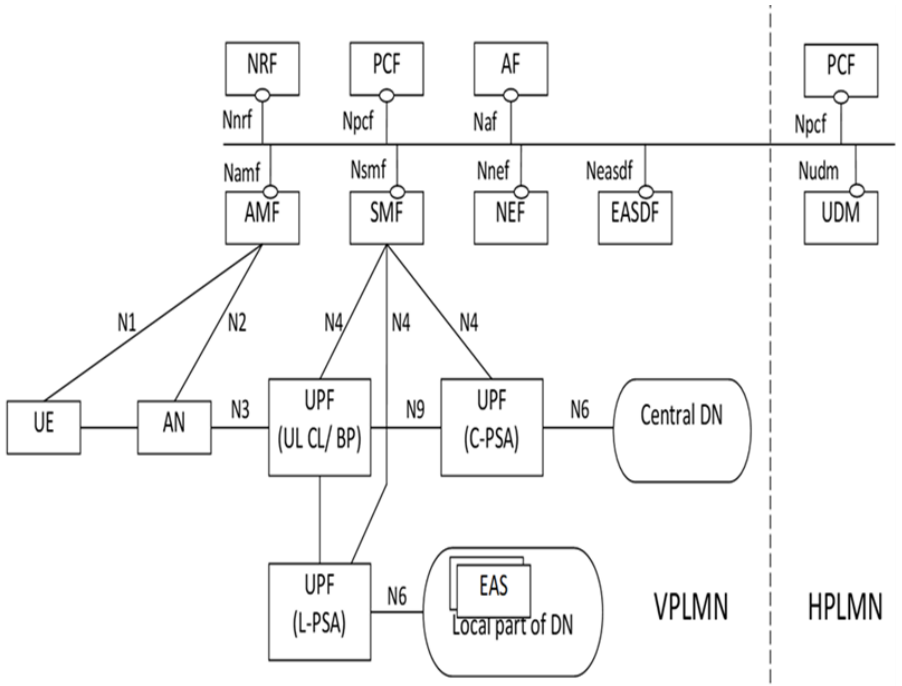


Fig.: 5G System providing access to Edge Application Server (EAS) with Data Traffic split to Local and/or Central DN scenario

2. LF Edge Akraino Contact for future inquiries and information

Asif Mahmood Awan
Ph.D. Computer Engineering at Jeju National University, Republic of Korea,

Akraino PTL for Akraino Blueprint: **MEC-based Stable Topology Prediction for Vehicular Networks**

E-mail: malikasifmahmoodawan@gmail.com