

ITU Focus Group Technical Specification

(10/2021)

Focus Group on Autonomous Networks

(FG-AN)

Use cases for autonomous networks



Technical Specification ITU FG-AN

Use cases for autonomous networks

Summary

This is a deliverable of the ITU-T Focus Group on Autonomous Networks (FG-AN).

This Technical Specification analyses use cases for autonomous networks. It provides use case descriptions and indicates the basic set of possible requirements for each use case. The use cases are divided into categories, priorities are indicated, and actor-interaction diagrams are added.

Keywords

Artificial Intelligence, autonomous networks, components, machine learning, requirements, use cases.

Note

This is an informative ITU-T publication. Mandatory provisions such as those found in ITU-T Recommendations are outside the scope of this publication. This publication should only be referenced bibliographically in ITU-T Recommendations.

Contributors:	Xiongwei Jia China Unicom China	Tel.: +86 15611092296 E-mail: jiawx9@chinaunicom.cn
	Abhay Shanker Verma TEC, Ministry of Communications India	E-mail: as.verma@gov.in
	Abhishek Dandekar Fraunhofer HHI Germany	E-mail: abhishek.girish.dandekar@hhi-extern.fraunhofer.de
	Abhishek Thakur Institute for Development and Research in Banking Technology (IDRBT) India	E-mail: AbhishekT@idrbt.ac.in
	Albert Cabellos- Aparicio Barcelona Neural Networking Universitat Politècnica de Catalunya Barcelona, Spain	E-mail: alberto.cabellos@upc.edu
	Albert López-Brescó Barcelona Neural Networking Universitat Politècnica de Catalunya Barcelona, Spain	E-mail: alopez@ac.upc.edu
	Alfons Mittermaier Highstreet Technologies GmbH Germany	E-mail: alfons.mittermaier@highstreet-technologies.com

Ammar Muthanna SPbSUT Russian Federation	E-mail: ammarexpress@gmail.com
Andrey Kucheryavy ITU-T SG11 Chairman	E-mail: akouch@mail.ru
Ashish S Sharma Fraunhofer HHI Germany	E-mail: ashish.sanjay.sharma@hhi.fraunhofer.de
Bing Qian China Telecom China	Tel: +8618511588556 E-mail: qianbing@chinatelecom.cn
Chi Wang Microsoft	E-mail: wang.chi@microsoft.com
Cleverson Veloso NAHUM Federal University of Pará Brazil	E-mail: cleversonahum@gmail.com
Dan Xu China Telecom China	Tel: +86 10 5090 2570 E-mail: xudan6@chinatelecom.cn
Gerhard Wieser FRINX Slovakia	E-mail: gwieser@frinx.io
Gokhan Kalem Turkcell Teknoloji Turkey	E-mail: gokhan.kalem@turkcell.com.tr
Guillaume Quintard	E-mail: guillaume.quintard@gmail.com
Jiaxin Wei China Unicom China	E-mail: weijx29@chinaunicom.cn
Jose Alcaraz Calero University of the West of Scotland (UWS) United Kingdom	E-mail: Jose.Alcaraz-Calero@uws.ac.uk
Julius Schulz-Zander Fraunhofer HHI Germany	E-mail: julius.schulz-zander@hhi.fraunhofer.de
Khakimov Abdukodir SPbSUT Russian Federation	E-mail: khakimov-aa@rudn.ru
Laurent Ciavaglia Rakuten Mobile Japan	E-mail: laurent.ciavaglia@rakuten.com
Liya Yuan ZTE Corporation China	E-mail: yuan.liya@zte.com.cn
Marco Gramaglia University Carlos III of Madrid Spain	E-mail: mgramagl@it.uc3m.es
Michaela Blott Xilinx United States	E-mail: mblott@xilinx.com

Michele Polese Northeastern University United States	E-mail: m.polese@northeastern.edu
Miquel Ferriol-Galmés Barcelona Neural Networking Universitat Politècnica de Catalunya Spain	E-mail: mferriol@ac.upc.edu
N. V. Narendra Kumar Institute for Development and Research in Banking Technology (IDRBT) India	E-mail: NVNarendra@idrbt.ac.in
Nik Sultana Illinois Institute of Technology United States	E-mail: nsultana1@iit.edu
Ozgur Ercetin Professor, Electronics Engineering Sabanci University Turkey	E-mail: oercetin@sabanciuniv.edu
Pål Hermunn Johansen Varnish Software Sweden	E-mail: hermunn@varnish-software.com
Paul Harvey Rakuten Mobile Japan	E-mail: paul.harvey@rakuten.com
Pradipta Biswas Indian Institute of Science (IISc) India	E-mail: pradipta@iisc.ac.in
Qi Sun China Mobile China	E-mail: sunqiyjy@chinamobile.com
Qi Wang University of the West of Scotland (UWS) United Kingdom	E-mail: Qi.Wang@uws.ac.uk
Ranjana Sivaram TEC, Ministry of Communications India	E-mail: ranjana.sivaram@gov.in
Salih Ergut Oredata Turkey	E-mail: salih.ergut@oredata.com
Shabnam Sultana Highstreet Technologies GmbH Germany	E-mail: shabnam.sultana@highstreet-technologies.com
Sirko Straube DFKI Germany	E-mail: sirko.straube@dfki.de

V. Udaya Sankar SRM University-AP India	E-mail: udayasankar.v@srmap.edu.in
Vijay Kumar Roy TEC, Ministry of Communications India	E-mail: vk.roy@gov.in
Vishnu Ram O.V. Independent Expert India	E-mail: vishnu.n@ieee.org
Wei Gao China Unicom China	E-mail: gaow25@chinaunicom.cn
Yongsheng Liu China Unicom China	E-mail: liuys170@chinaunicom.cn

This Technical Specification has been published as approved by the focus group, without any subsequent editorial review.

© ITU 2026

Some rights reserved. This publication is available under the Creative Commons Attribution-Non Commercial-Share Alike 3.0 IGO licence (CC BY-NC-SA 3.0 IGO; <https://creativecommons.org/licenses/by-nc-sa/3.0/igo>).

If you wish to reuse material from this publication that is attributed to a third party, such as tables, figures or images, it is your responsibility to determine whether permission is needed for that reuse and to obtain permission from the copyright holder. The risk of claims resulting from infringement of any third-party owned material in the publication rests solely with the user.

Table of Contents

	Page
1 Scope.....	1
2 References.....	1
3 Definitions	1
3.1 Terms defined elsewhere	1
3.2 Terms defined in this Technical Specification	1
4 Abbreviations and acronyms	2
5 Conventions	3
6 Introduction.....	3
7 Use cases.....	4
7.1 Import and export of knowledge for autonomous network	4
7.2 Configuring and driving simulators from autonomous components in the network	6
7.3 Peer-in-loop (including humans).....	8
7.4 Configuring and driving automation loops from autonomous components in the network	9
7.5 Domain analytics services for E2E service management.....	12
7.6 Automation and intelligent operation, maintenance and management (OAM) of radio network	13
7.7 Intelligent energy saving for data centres.....	14
7.8 Autonomous massive MIMO	15
7.9 Network resource allocation for emergency management based on closed loop analysis	16
7.10 Inter-domain service automation (IDSA) – for microfinance	20
7.11 Autonomous vertical-driven edge service and middle-mile connectivity for rural financial inclusion (FI)	23
7.12 Signalling flows for autonomous IMT-2020 network.....	25
7.13 Plug/play of network instance	26
7.14 "Generative adversarial Sandbox": (or hybrid closed loops)	27
7.15 Open, integrated, log analysis	28
7.16 Compose-able, hierarchical closed loops	29
7.17 Quality of Experience (QoE) Prediction as-a-Service (QPaaS)	30
7.18 Autonomy applied to CDNs	32
7.19 Analysis-driven evolution in virtualized RAN based on devops	34
7.20 Evolving edge applications for verticals using private 5G.....	36
7.21 Experimentation and "fire-drills" for public safety networks	38
7.22 Machine learning for network automation	39
7.23 Autonomous agents (with varied competence) in networks.....	41
7.24 Automated, adaptive acceleration for AI @ edge	43

7.25	Assistive networks: Adaptation of communication system based on changing user accessibility needs	45
7.26	Ev-as-a-service: Achieving zero touch evolution in a delegated autonomy case	47
7.27	Experimentation as a service: Digital twins as platforms for experimentation	49
7.28	Evolution from scenario-specific, explicit-coordination to coordination-free interoperability (achieved using data-driven approaches)	51
7.29	Intelligent maintenance assistance system	53
7.30	Demand forecasting and live service migration methods in edge computing systems	56
7.31	OpenCN: An open repository of intents for controllers and modules.....	58
7.32	AI enabled game theory-based mechanism for resource allocation	60
7.33	Service automation using workflows	65
7.34	Disaggregation and placement of in-network programs	68
7.35	A fast and lightweight autoML library (FLAML).....	70
7.36	Connected AI (CAI) testbed: Testbed for 5G connected artificial intelligence on virtualized networks.....	72
7.37	Negotiated boundaries in AN for seamless network sharing	74
7.38	AN enabled end-to-end supply chain	77
7.39	Towards openness in AN.....	80
7.40	Awareness in AN.....	81
	Bibliography.....	85

Technical Specification

Use cases for Autonomous Networks

1 Scope

This Technical Specification analyses use cases for autonomous networks. It provides use case descriptions and indicates the basic set of possible requirements for each use case. The use cases are divided into categories, priorities are indicated, and actor-interaction diagrams are added.

2 References

- [ITU-T Y.3172] Recommendation ITU-T Y.3172 (2019), *Architectural framework for machine learning in future networks including IMT-2020*.
- [ITU-T Y.3173] Recommendation ITU-T Y.3173 (2020), *Framework for evaluating intelligence levels of future networks including IMT-2020*.
- [ITU-T Y.3174] Recommendation ITU-T Y.3174 (2020), *Framework for data handling to enable machine learning in future networks including IMT-2020*.
- [ITU-T Y.3176] Recommendation ITU-T Y.3176 (2020), *Machine learning marketplace integration in future networks including IMT-2020*.
- [ITU-T Y.3179] Recommendation ITU-T Y.3179 (2021), *Architectural framework for machine learning model serving in future networks including IMT-2020*.

3 Definitions

3.1 Terms defined elsewhere

This Technical Specification uses the following terms defined elsewhere:

3.1.1 application service [b-ITU-T X.1121]: A service like mobile banking, mobile commerce, and so on.

3.1.2 network service [b-ITU-T Y.3515]: A collection of network functions with a well specified behaviour.

3.2 Terms defined in this Technical Specification

This Technical Specification defines the following terms:

3.2.1 autonomous network (AN) components: Logical functions which work together to achieve autonomous behaviour including evolution, exploration and adaptation.

NOTE – Examples of AN components are knowledge base and AN Sandbox. There may be other components enabling evolution, exploration and adaptation.

3.2.2 autonomous network (AN) configuration: A set of parameters which are input to the AN, which may control its behaviour.

3.2.3 autonomous network (AN) sandbox: An AN component which hosts simulators (which act as data generators and provide action sinks), provide data handling interfaces to real network. This component is based on ML Sandbox [ITU-T Y.3172], but broader to include all types of AN experimentation as against only AI/ML model testing and validation.

3.2.4 industry vertical applications: Software realization of a workflow created by application services [b-ITU-T X.1121], to serve their customers' specific needs.

NOTE – e.g., a set of cloud native services which implement loan management in banking industry vertical.

3.2.5 industry vertical solution provider: A domain specific business enterprise with specific customer needs, a part of which may need the services of an ICT network.

NOTE – e.g., a bank which offers financial services to its customers and uses a 5G network to connect to its customers.

3.2.6 knowledge: A collection of resources that helps in solving a specific type of problem.

NOTE – Examples of resources are description of a problem along with the description of corresponding potential solutions to that type of problem. The descriptions may be in the form of standard metadata. Resources may include possible causes of the problem, corresponding solutions and their advantages, disadvantages and optimization approaches etc. Problems may have sub-problems e.g., QoE problems may have sub-problems including coverage problems and/or interference problems.

3.2.7 knowledge base: A logical collection of knowledge that are related in a specific manner.

NOTE – Examples of logical collection are grouping of knowledge resources with labels specifying their common relationship, graph of knowledge resources with edges representing the relationships, etc. Knowledge base may be used to solve problems in a particular domain such as access networks or core networks, etc.

4 Abbreviations and acronyms

This Technical Specification uses the following abbreviations and acronyms:

AI	Artificial Intelligence
AN	Autonomous Networks
CI/CD	Continuous Integration and Continuous Delivery
CN	Controller
ER	Emergency Response
GNN	Graph Neural Networks
GUI	Graphical User Interface
IDSA	Inter-domain Service Automation
KB	Knowledge Base
KPI	Key Performance Indicator
LCM	Life Cycle Management
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MLFO	Machine Learning Function Orchestrator
mMTC	Massive Machine Type Communications
MNO	Mobile Network Operator
NF	Network Function
nRT RIC	Near Real Time RIC
OSS	Operational Support System
QoE	Quality of Experience
QoS	Quality of Service
QPaaS	Quality of Experience (QoE) Prediction as-a-Service
RAN	Radio Access Network

RIC	RAN intelligent Controller
RSRP	Reference Signal Received Power
SINR	signal-to-interference-plus-noise ratio
SLA	Service Level Agreement
TOSCA	Topology and Orchestration Specification for Cloud Applications
URLLC	Ultra-reliable low-latency communication
ZSM	Zero Touch Service Management

5 Conventions

In this Technical Specification, in alignment with the conventions of Supplement 55 to ITU-T Y-series Recommendations possible requirements which are derived from a given use case, are classified as follows:

The keywords "it is critical" indicate a possible requirement which would be necessary to be fulfilled (e.g., by an implementation) and enabled to provide the benefits of the use case.

The keywords "it is expected" indicate a possible requirement which would be important but not absolutely necessary to be fulfilled (e.g., by an implementation). Thus, this possible requirement would not need to be enabled to provide complete benefits of the use case.

The keywords "it is of added value" indicate a possible requirement which would be optional to be fulfilled (e.g., by an implementation), without implying any sense of importance regarding its fulfilment. Thus, this possible requirement would not need to be enabled to provide complete benefits of the use case.

6 Introduction

As the demand and expectation of communication networks has grown, so have user subscription and new service expectation. Network operators must find new ways to address these pressures while at the same time controlling operational cost. Autonomous networks are those that possess the ability to monitor, operate, recover, heal, protect, optimize, and reconfigure themselves; these are commonly known as the self-* properties. The impact of autonomy on the network will be in all areas including planning, security, audit, inventory, optimisation, orchestration, and quality of experience. In this context, the main concepts studied by FG AN are exploratory evolution, real-time responsive experimentation and dynamic adaptation.

Use cases studied in this document are based on contributions and discussions with domain experts or mentors. There are various types of use cases including those which are directly describing various autonomous behaviours and those which describe the applications which benefit from them. Collation of use cases included specific effort to study the impacts to the key concepts under study in the FG AN. Effort was made to derive requirements and further to classify the requirements. A relation to the architecture is provided in the form of guidance to components derived from the use cases.

The main learnings from this use case analysis are:

- While the use cases for AN are quite varied and requires support from domain experts, a common refrain has been the application of the key concepts mentioned above.
- Use case analysis may need to be continued as the field is still evolving.
- Derivation of architecture concepts and proof of concepts from the use cases is important.

7 Use cases

7.1 Import and export of knowledge for autonomous network

<i>Use case id</i>	FG-AN-usecase-001
<i>Use case name</i>	Use of knowledge in autonomous network
<i>Base contribution</i>	[FGAN-I-12-R1]
<i>Creation date</i>	21/January/2021
<i>Use case context</i>	Discussions during ITU webinar on autonomous networks (3 November 2020)
<i>Use case description</i>	<p>To satisfy the key concepts of autonomous networks (evolution, experimentation and adaptation) while minimizing human intervention requires knowledge. This knowledge may include representation of data about the environment in which the autonomous system is operating, possible actions and consequences, key configuration options, potential parameter indices and other types of logic. This use case concerns use of knowledge in the actors in the AN.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Knowledge is imported from outside or peer entities of autonomous network components 2. Knowledge is referred internally in autonomous network components e.g., for driving evolution, driving exploration, configuration of automation loops etc. 3. Generate report for human consumption 4. Knowledge is stored and updated within the autonomous network components 5. Knowledge is exported from autonomous network components to outside or peer entities.
<i>Open issues</i>	<ul style="list-style-type: none"> – Representation mechanisms and transfer protocols for knowledge. – Capabilities of knowledge base (KB) component that allows storage, query, export, import and modification of knowledge using standard mechanisms is under study. – Reference points and interfaces between AN and KB is yet to defined.
<i>Use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Reference</i>	[b-Clark], [b-AN2020], [b-Jimenez-Ruiz], [b-Myklebust], [b-Turing]

7.1.1 Use case requirements

Critical requirements

- AN-UC01-REQ-001: It is critical that AN enable exchange of knowledge between the various components in the AN implementing the various key AN functionalities like evolution, exploration and adaptation.
- AN-UC01-REQ-002: It is critical that AN enable optimization of knowledge bases.

NOTE – Examples of optimizations applied on the knowledge bases are access policies, granularity of storage, interconnection between various knowledge bases and relation between problems and solutions, addition of new knowledge.

- AN-UC01-REQ-003: It is critical that AN enable creation of reports on the use of knowledge bases, for consumption of humans and machines.

NOTE – Example of contents of reports are metrics on access by various components in the AN, other network services in and outside its administrative domains.

- AN-UC01-REQ-004: It is critical that AN enable exchange of knowledge between various components in the AN, and other network services in the same administrative domain.
- AN-UC01-REQ-005: It is critical that AN use knowledge base for mapping high level use case description to controller description.

NOTE – Controller description may use languages such as TOSCA, whereas use case descriptions may be unstructured. The high-level use case description is to be converted to a structured controller specification, also known as Intent, by AN orchestrator. In this process of "conversion", it may utilize the help of humans (using GUIs) who can better understand unstructured information, and/or auto-controller generators.

Expected requirements

- AN-UC01-REQ-006: It is expected that AN enable exchange of knowledge between various components in the AN, and entities in other administrative domains.

NOTE – Examples of entities in other administrative domains are network services which are not implementing AN functionalities.

Added value requirements

- AN-UC01-REQ-007: It is of added value that AN use Auto-controller generators that are functions which generate controller specifications, using the existing repository in openCN, the knowledge base and an analytics function aided by AI/ML e.g., GNN, recommendation engine.

7.1.2 Use case specific figures

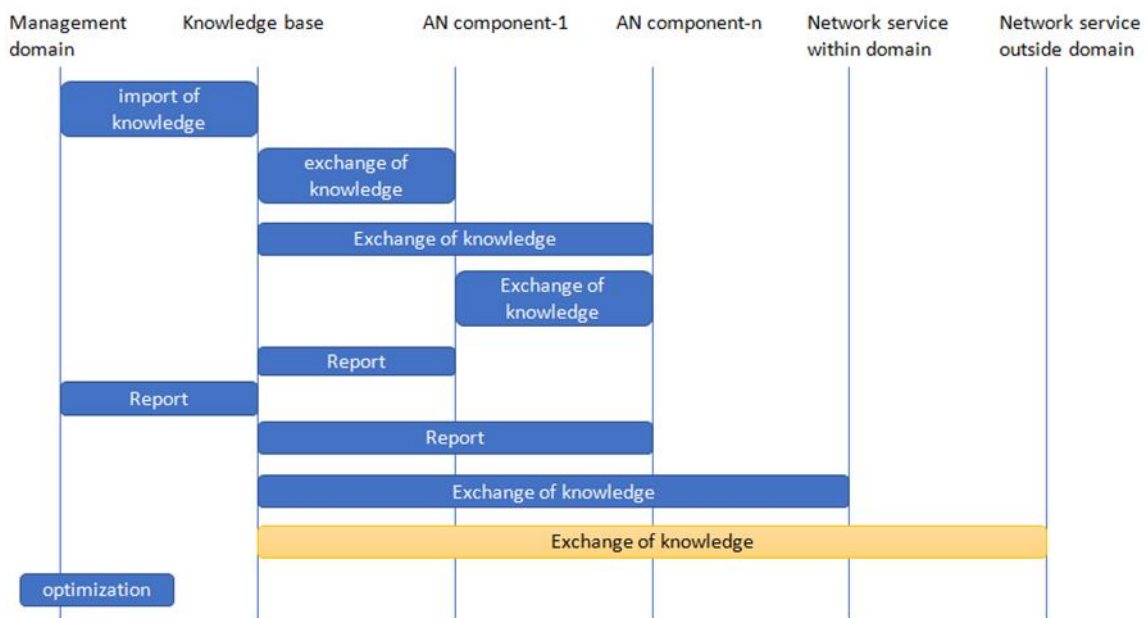


Figure 1 – Actor interaction for Use of knowledge in autonomous network

NOTE 1 – Multiple AN components (1 ... n) may be present in the system.

NOTE 2 – Actor interaction captured in this figure may not reflect the strict time sequence of activities.

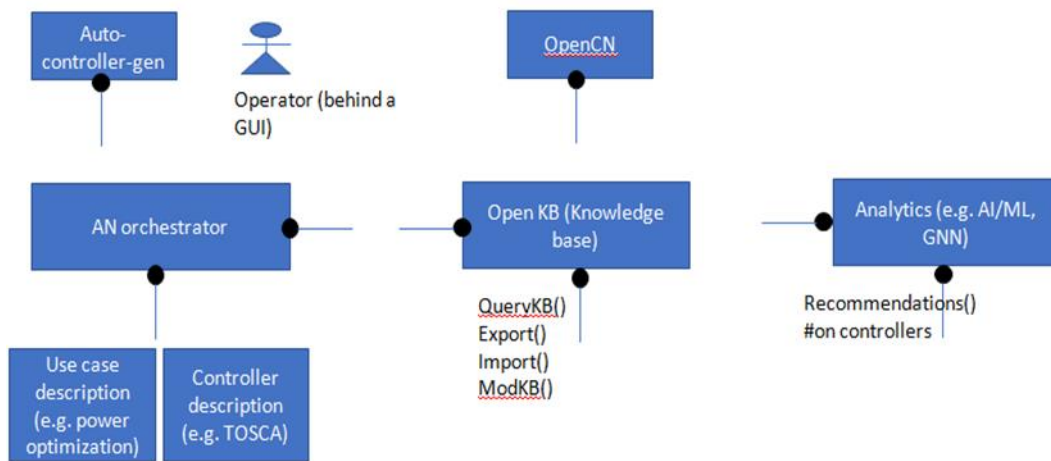


Figure 2 – Component cloud for Use of knowledge in autonomous network

7.2 Configuring and driving simulators from autonomous components in the network

<i>Use case id</i>	FG-AN-usecase-002
<i>Use case name</i>	Configuring and driving simulators from autonomous components in the network
<i>Base contribution</i>	[FGAN-I-12-R1]
<i>Creation date</i>	21/January/2021
<i>Use case context</i>	Discussions during ITU workshop on autonomous networks, discussions during review of ITU SG 13 work item on Sandbox, discussions during ITU AI/ML in 5G Challenge 2020.
<i>Use case description</i>	<p>To explore and experiment with various scenarios in autonomous behaviour, the autonomous network component requires access to simulators. Simulators help evaluate the outcome of possible options without potential adverse fallouts in the real network. Long term study of simulation results is common by human researchers to understand the evolutionary needs of the network too. In this respect, autonomous network components need to interface with, configure and drive the different simulators.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Autonomous network components decide the scenarios for exploration and experimentation. 2. Autonomous network components interact with the Sandbox to configure specific simulators which can perform the required experimentation. 3. Sandbox monitors the simulators and reports the completion of simulations. 4. The results are analysed by Autonomous network components and further actions (like updating knowledge base) are taken.
<i>Open issues</i>	<ol style="list-style-type: none"> 1. Are simulators encapsulated in Sandbox? Or are they open to direct interface from autonomous network components? 2. There are heterogeneous simulators and uniform interface with simulators do not exist. This makes their interface and configuration non-standard and difficult to implement. 3. Additional scenarios like addition of new simulation capabilities, flagging of new requirement for simulation etc. need to be handled.
<i>Use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Reference</i>	[b-Y.ML-IMT2020-SANDBOX]

7.2.1 Use case requirements

Critical requirements

- AN-UC02-REQ-001: It is critical that AN components arrive at potential scenarios for exploration and experimentation.

NOTE – AN components may independently arrive at different scenarios for exploration and experimentation based on several factors like the functionalities they implement, current status of their knowledge, etc. e.g., exploration strategies for access control may be based on game theory approaches or combinatorial optimization approaches.

- AN-UC02-REQ-002: It is critical that autonomous networks (AN) components trigger experimentation in AN Sandbox.

NOTE – AN components may independently trigger experimentation by configuring simulators in the AN Sandbox.

- AN-UC02-REQ-003: It is critical that autonomous networks (AN) Sandbox collate, aggregate triggers for experimentation to form a coherent, experimentation pipeline, the execution of which is monitored and reported by AN Sandbox to AN components.
- AN-UC02-REQ-004: It is critical that autonomous networks (AN) components analyse the reports from the AN sandbox while considering the steps in AN behaviour.

NOTE – The steps in AN behaviour which depends on the analysis of reports from AN sandbox includes steps in evolution and update of knowledge.

7.2.2 Use case specific figures

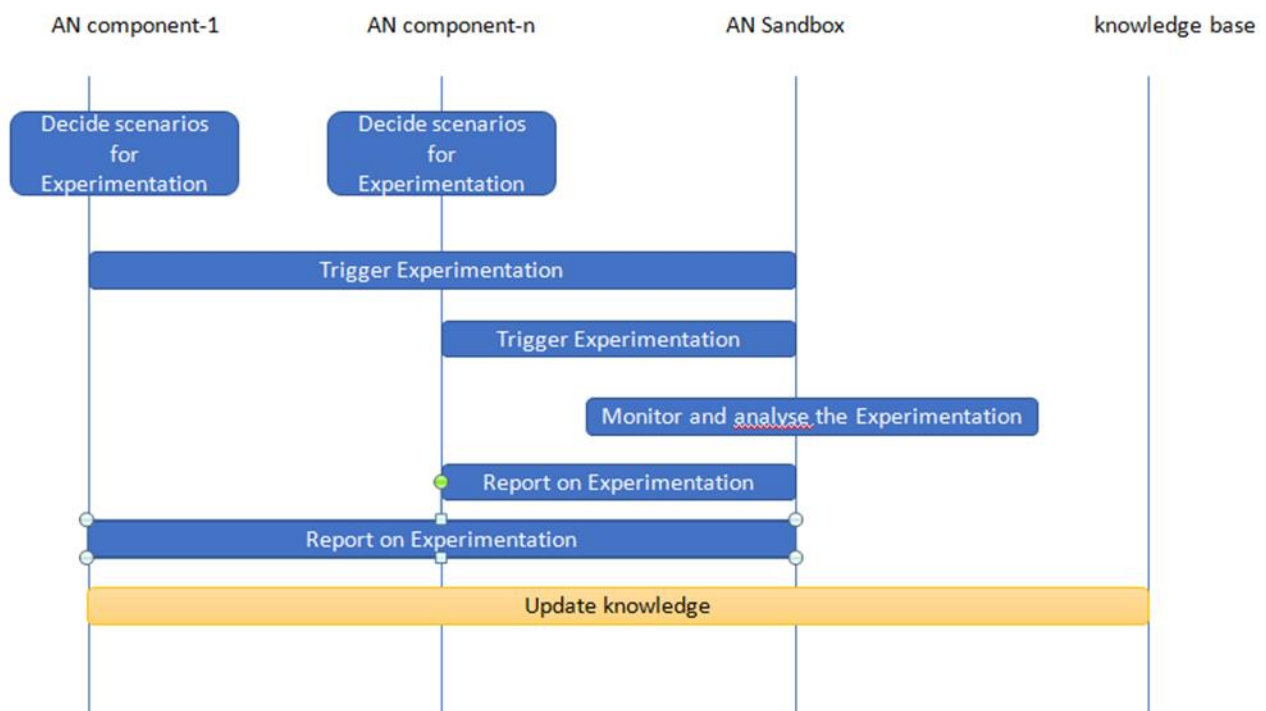


Figure 3 – Actor interaction for Configuring and driving simulators from autonomous components in the network

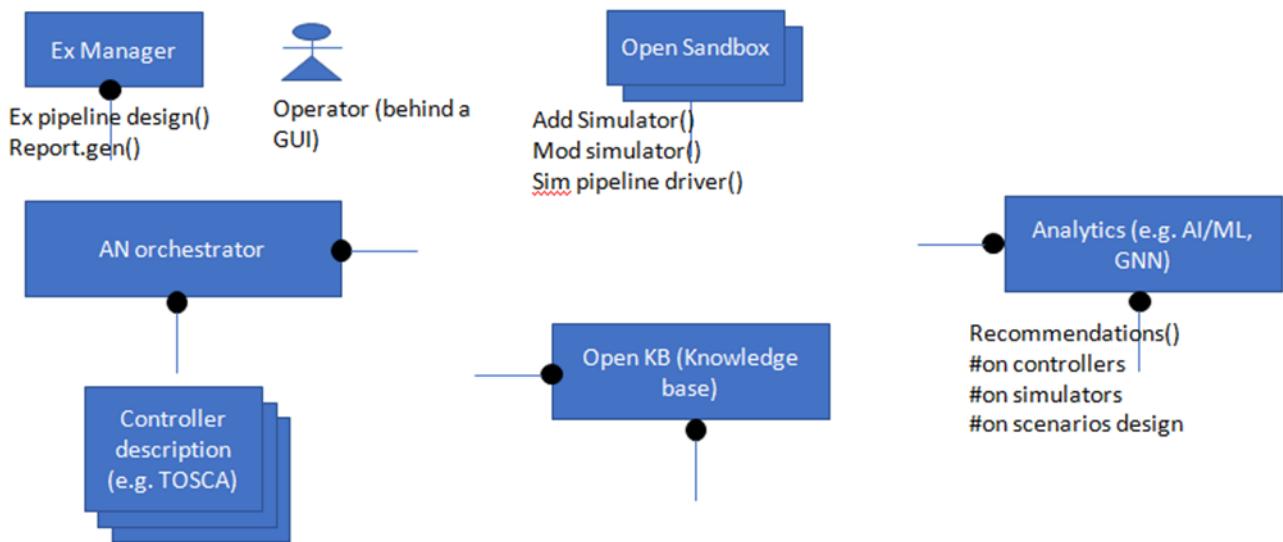


Figure 4 – Component cloud for Configuring and driving simulators from autonomous components in the network

7.3 Peer-in-loop (including humans)

<i>Use case id</i>	FG-AN-usecase-003
<i>Use case name</i>	Peer-in-loop (including humans)
<i>Base contribution</i>	[FGAN-I-12-R1]
<i>Creation date</i>	21/January/2021
<i>Use case context</i>	Discussions during ITU workshop on autonomous networks, discussions during review of ITU SG 13 work item on MLFO, discussions during ITU AI/ML in 5G Challenge 2020.
<i>Use case description</i>	To guide the autonomous behaviour autonomous network component requires access to peers. Peers include humans and other autonomous entities. Exchange of information with peers help in taking better decisions. In this respect, autonomous network components need to interface with, exchange information with various other autonomous network components and humans. Following are related steps in this use case scenario: 1. Autonomous network components decide to take guidance from other autonomous entities (peers like humans). 2. A message exchange with the peer is initiated. 3. The results of the exchange are analysed by Autonomous network components and further actions (like updating knowledge base) are taken.
<i>Open issues</i>	1. Are some peers more equal than others (e.g., humans)? 2. What are the messages exchanged? e.g., request for comments? report on status? capability exchange?
<i>Use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Reference</i>	[b-Y.ML-IMT2020-MLFO]

7.3.1 Use case requirements

Critical requirements

- AN-UC03-REQ-001: It is critical that autonomous networks (AN) components enable synchronous or asynchronous, interoperable exchange of feedback or information from peers regarding the decisions and choices related to AN behaviour.

NOTE 1 – Peers may include humans and machines. Feedback may include exchange of information regarding AN behaviour such as evolution, experimentation and adaptation. Contents of the information exchanged may include capabilities and status of components in the AN e.g., knowledge base, orchestration, simulators, etc.

NOTE 2 – Format of information exchange is for future study.

7.3.2 Use case specific figures

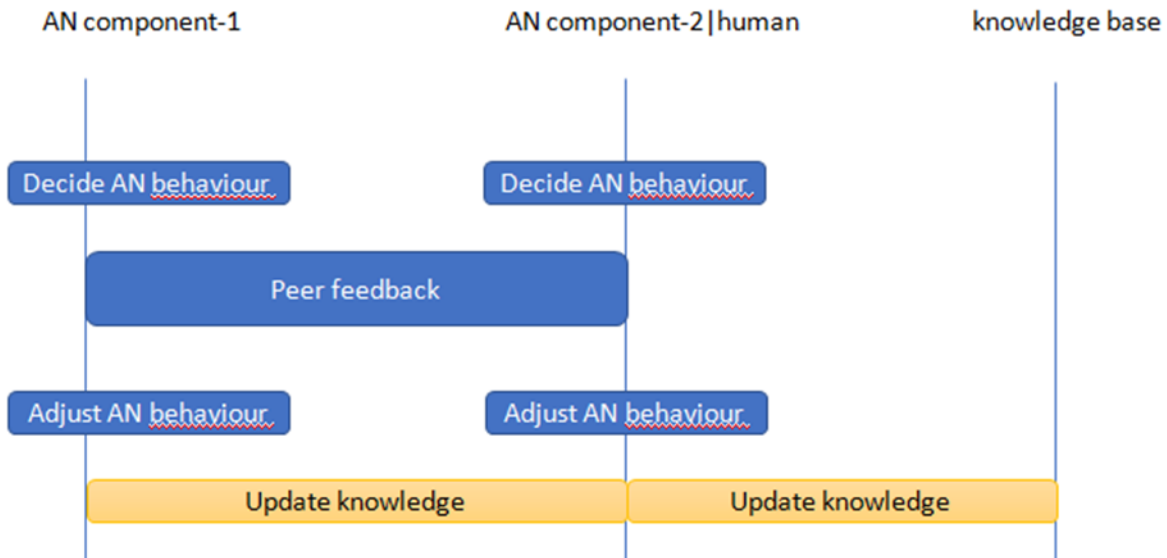


Figure 5 – Actor interaction for Peer-in-loop (including humans)

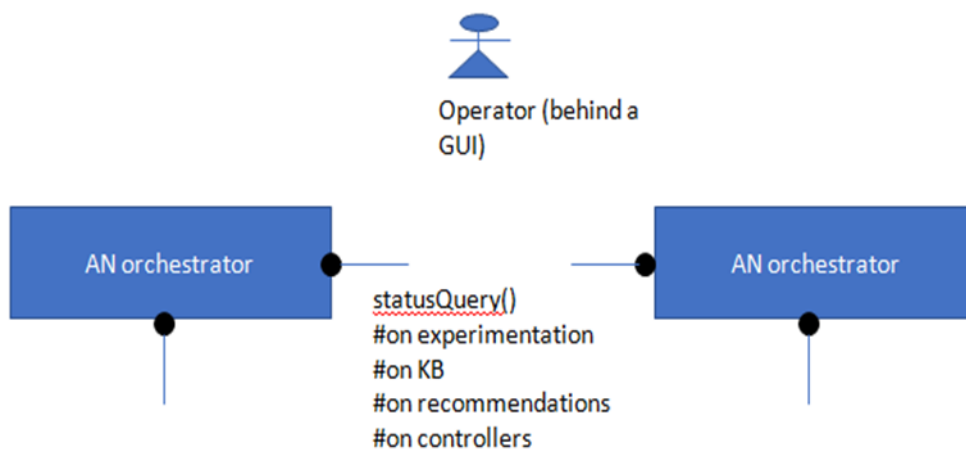


Figure 6 – Component cloud for Peer-in-loop (including humans)

7.4 Configuring and driving automation loops from autonomous components in the network

<i>Use case id</i>	FG-AN-usecase-004
<i>Use case name</i>	Configuring and driving automation loops from autonomous components in the network
<i>Base contribution</i>	[FGAN-I-12-R1]
<i>Creation date</i>	21/January/2021
<i>Use case context</i>	Inspired by discussions on "demand mapping" during Y.3173 and discussions during ITU AI/ML in 5G Challenge 2020.

<i>Use case description</i>	<p>There are different automation loops in various domains of the network, already proposed by different standards bodies and industry bodies. To reflect the decisions of autonomous behaviour in the network, the autonomous network component requires access to automation loops. Automation loops help implement the decisions taken by the autonomous component in the network. Moreover, it is possible that automation loops provide valuable inputs for autonomous components to be considered for say further experimentation.</p> <p>In this respect, autonomous network components need to interface with, configure and drive the different automation loops.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Autonomous network components decide the configurations of automation loops. 2. Autonomous network components interact with the automation loops to configure specific scenarios which can perform the required automation. 3. Automation loops monitor the automation and reports the status of automation. 4. The results are analysed by Autonomous network components and further actions (like updating knowledge base) are taken. <p>NOTE – Please see [ML5G-I-221] for an example of "influencing analytics service" for E2E service management. Question-8 discussed the possibility for ZSM framework consumer to select and provision the type of ML model to be used for domain specific analytics in ZSM scope.</p>
<i>Open issues</i>	<ol style="list-style-type: none"> 1. Where are the automation loops hosted? Are they open to direct interface from autonomous network components? 2. There are heterogeneous automation loops and uniform interface do not exist. This makes their interface and configuration non-standard and difficult to implement. 3. Additional scenarios like addition of new automation capabilities, flagging of new requirement for automation etc. need to be handled.
<i>Use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Reference</i>	[ITU-T Y.3173]

7.4.1 Use case requirements

Critical requirements

- AN-UC04-REQ-001: It is critical that autonomous networks (AN) components decide the type of closed loops and manage the closed loops.

NOTE – AN components may decide the type and structure of closed loops based on their analysis of reports, monitoring and other information exchanges. Management of closed loops may include instantiating, deletion, updating, and other operations on closed loops.

- AN-UC04-REQ-002: It is critical that autonomous networks (AN) components consider the capability and flexibility offered by closed loops to configure them to perform specific automation tasks.
- AN-UC04-REQ-003: It is critical that closed loops monitor the specific parameters of automation tasks and report them to AN components.

NOTE – Specific parameters of automation tasks may include data input to automation, analytics used in the closed loop, actions taken as part of automation. It may also include failures, error logs, etc.

- AN-UC04-REQ-004: It is critical that AN components consider the reports from closed loops while deciding the AN behaviour.

NOTE – Examples of AN behaviour are evolution, experimentation and adaptation.

7.4.2 Use case specific figures

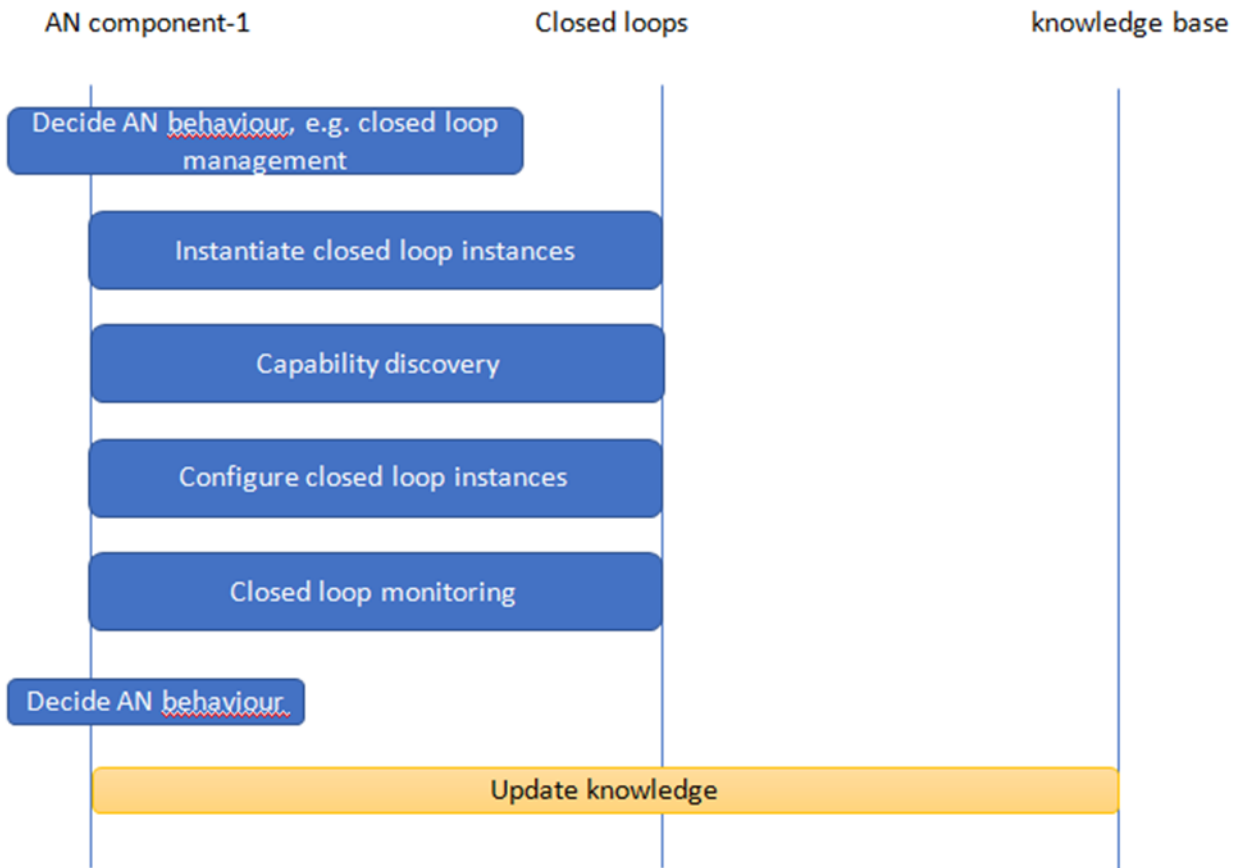


Figure 7 – Actor interaction for Configuring and driving automation loops from autonomous components in the network

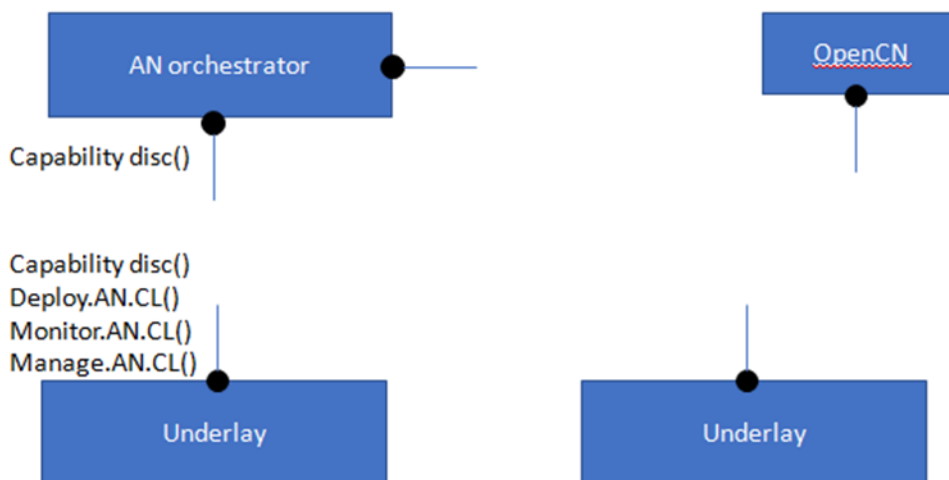


Figure 8 – Component cloud for Configuring and driving automation loops from autonomous components in the network

7.5 Domain analytics services for E2E service management

<i>Use case id</i>	FG-AN-usecase-005
<i>Use case name</i>	Domain analytics services for E2E service management
<i>Base contribution</i>	[FGAN-I-12-R1]
<i>Creation date</i>	21/January/2021
<i>Use case context</i>	Based on discussions with ETSI ZSM via [ML5G-I-221] Based on presentations on [FGAN-I-135].
<i>Use case description</i>	Section 6.5.3.2 of [ETSI ZSM ARCH] describes the domain analytics services which provide domain-specific insights and generate domain-specific predictions based on data collected by domain data collection services and other data. Following are related steps in this use case scenario: <ol style="list-style-type: none"> 1. Autonomous network components act as a ZSM service consumer. 2. ZSM provides closed loop (CL) management and other domain and cross domain services (including analytics) to AN components. 3. Discovery of ZSM services is done by AN components 4. ZSM service performs the E2E service management based on the interaction with autonomous network component. <ol style="list-style-type: none"> a. examples of interactions are: managing subscriptions, configuring analytics, request analysis results, etc. See clause 6.5.3.2.1 of [ETSI ZSM ARCH] b. Other Examples in the context of Zero Touch provisioning mentioned in [FGAN-I-135].
<i>Open issues</i>	Open issues (refer also those discussed in [ML5G-I-221]). <ol style="list-style-type: none"> 1. Hierarchies of CL may be deployed in other AN domains and ZSM domains. In this case interoperability of these deployments may be studied. 2. How to integrate ZSM services as "service-x"?
<i>Use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Reference</i>	[b-ETSI GS ZSM 002]

7.5.1 Use case requirements

Critical requirements

- AN-UC05-REQ-001: It is critical that AN support discovery and consumption of the services provided by different types of closed loop service automation frameworks.

NOTE – Examples of different types of closed loop service automation frameworks are ETSI ZSM [ETSI ZSM ARCH] and FRINX machine [FGAN-I-135]. Examples of actions taken by AN after the consumption of services provided by closed loop service automation frameworks are managing subscriptions, configuring analytics, request analysis results, etc.

7.5.2 Use case specific figures

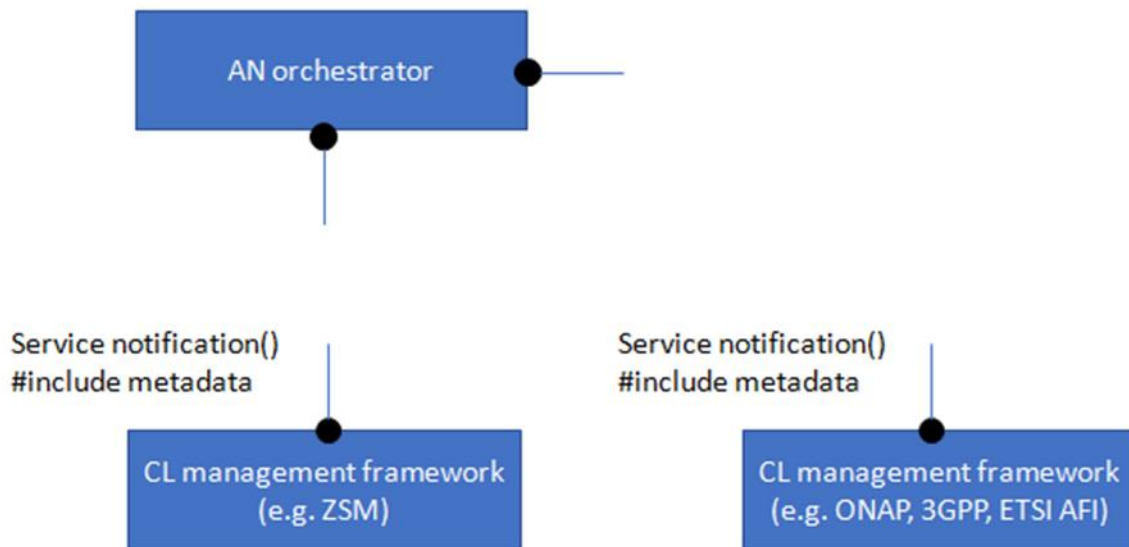


Figure 9 – Component cloud for Domain analytics services for E2E service management

7.6 Automation and intelligent operation, maintenance and management (OAM) of radio network

<i>Use case id</i>	FG-AN-usecase-006
<i>Use case name</i>	Automation and intelligent OAM (operation, maintenance and management) of radio network
<i>Base contribution</i>	[FGAN-I-008]
<i>Description</i>	<p>Background: Dynamic radio environment, network structure, user behaviour, and user distribution drive the network needs to be monitored and optimised continually. Currently, it consumes experts a lot of time and labor work to discover problems, analyse root cause, and then formulating solutions of radio networks. Therefore, AI and big data technology is necessary to achieve full process automation and intelligent management of wireless network.</p> <p>Specifically, wireless network autonomous management includes real-time monitoring of data quality, quasi-real-time diagnosis of abnormal cells, root cause analysis, recommended solutions, and evaluation of processing effects.</p> <ul style="list-style-type: none"> • Real-time monitoring of data quality <p>It is needed to firstly collect real-time data from the 4/5G integrated network management, then compares the consistency of the number of network elements from the data, and achieves data quality monitoring and warning through the visualization panel.</p> <ul style="list-style-type: none"> • Quasi-real-time diagnosis of abnormal cells <p>Using several categories of KPI performance indicators, the ML algorithm (e.g., OneClassSVM) is used to diagnose network elements in these categories of scenarios such as residential and colleges on a daily/weekly basis, and distribute them to frontline managers according to the abnormal probability.</p> <ul style="list-style-type: none"> • Root cause analysis and recommended solutions <p>Through collecting tens of thousands of expert experiences, and radio network knowledge bases have been established through knowledge graphs to develop intelligent recommendation algorithms and search engines, and to</p>

	<p>directly provide reasons and recommend solutions of each abnormal network element to first-line experts, thus reducing troubleshooting time and error rate.</p> <ul style="list-style-type: none"> • Evaluation of processing effects <p>Through a mature evaluation system, the effectiveness of the solution to each abnormal problem is evaluated after the implementation, and then the knowledge base and recommendation algorithm are optimized, and the intelligence level of the entire process is continuously improved.</p>
<i>Category</i>	Cat 1: Use case for autonomous behaviour.
<i>Reference</i>	None

7.6.1 Use case requirements

Critical requirements

- AN-UC06-REQ-001: It is critical that autonomous networks enable discovery of problems in underlay networks, analysis of root cause, and formulation of solutions.

NOTE – Wireless network autonomous management may include real-time monitoring of data quality, quasi-real-time diagnosis of abnormal cells, root cause analysis, recommended solutions, and evaluation of processing effects.

- AN-UC06-REQ-002: It is critical that AN enables data quality and monitoring and visualization.

NOTE – Data quality may need real-time monitoring, evaluation with respect to openKB, and reporting may be done using an online GUI or a report to human. Data quality may be useful to analytics services.

- AN-UC06-REQ-003: It is critical that AN enables capturing and using the knowledge from domain experts and AI/ML mechanisms for recommendation of solution for root cause analysis.

NOTE – Example of representation formats for knowledge is knowledge graphs.

Expected requirements

- AN-UC06-REQ-004: It is expected that AN uses AI and big data technology to achieve full process automation and intelligent management of wireless network.

Added value requirements

- AN-UC06-REQ-005: It is of added value that a varying set of KPIs are monitored to identify faults.
- AN-UC06-REQ-006: It is of added value that AN solutions may be monitored optimized and continuously improved (themselves).

NOTE – e.g., the OpenKB and recommendation algorithms may be optimized.

7.6.2 Use case specific figures

None.

7.7 Intelligent energy saving for data centres

<i>Use case id</i>	FG-AN-usecase-007
<i>Use case name</i>	Intelligent Energy Saving for Data Centers
<i>Base contribution</i>	[FGAN-I-008]
<i>Description</i>	Background: The rapid growth of mobile Internet, cloud computing and other business drives the need of large-scale data centers. Data centers consume large amounts of energy to run and maintain their cooling system and facilities, servers and other devices. Traditional methods cannot efficiently reduce the energy costs of data centers. Therefore, AI

	mechanisms are introduced to analyse the monitoring data and adjust the configurations automatically. Intelligent energy saving solution include a series of autonomous behaviour, such as automatic data acquisition, AI-based energy consumption modelling and inference, facilities parameters control policies decision, facilities adjustment actions implementation, energy saving result evaluation and control policies continuous optimization.
<i>Category</i>	Cat 2: Application of autonomous behaviour.
<i>Reference</i>	None

7.7.1 Use case requirements

Critical requirements

- AN-UC07-REQ-001: It is critical that autonomous networks (AN) support data acquisition, representation, analysis of collected data and adaptation of configurations in underlays such as data centers.

NOTE – Data acquisition and data representations may use industry standards. Analysis may use ML techniques. Adaptations may use underlay specific APIs. Adaptations may be arrived at using controllers or workflows or closed loops.

Expected requirements

- AN-UC07-REQ-002: It is expected that autonomous networks (AN) support representation, autonomous analysis and continuous optimization of policies.

NOTE – Policies may be related to domain specific workflows and decisions e.g., energy usage in data centers.

7.7.2 Use case specific figures

None.

7.8 Autonomous massive MIMO

<i>Use case id</i>	FG-AN-usecase-008
<i>Use case name</i>	Autonomous Massive MIMO
<i>Base contribution</i>	FGAN-I-30
<i>Creation date</i>	27/January/2021
<i>Use case context</i>	
<i>Use case description</i>	<p>Massive MIMO is a key technology in 5G, which can effectively improve the vertical coverage and system capacity in complex scenarios by using large-scale antenna array and three-dimensional beam-forming.</p> <p>Compared with traditional antenna, there are more dimensions of parameters to adjust for massive MIMO large-scale antenna array, including horizontal lobe width, vertical lobe width, azimuth, dip angle and beam number. Each dimension can be fine adjusted by setting a reasonable step size and theoretically and there may be tens of thousands possible combination of antenna parameter weights in a cell. Therefore, manual optimization and adjustment based on scenario/service changes can be very hard in consideration of multi-cell coordination.</p> <p>The autonomous massive MIMO use case is about helping operators quickly converge and achieve optimal adjustment of antenna parameters with AI capabilities of multi-dimensional analysis and prediction. The general workflow is: 5G base station collects position information from UE and sends it to the network management system, which then calculates the distribution of UE and finds the optimal weight combination with ML algorithms based on</p>

	the target RSRP/SINR distribution in the current scenario, so as to maximize the utilization of system capacity and guarantee the user experience.
<i>Open issues</i>	<ul style="list-style-type: none"> • How to achieve a balance between coverage, communication traffic and spectrum efficiency? • How to ensure a satisfactory speed of parameter adjustment?
<i>Use case category</i>	Cat 2: Application of autonomous behaviour.
<i>Reference</i>	

7.8.1 Use case requirements

Critical requirements

- AN-UC08-REQ-001: It is critical that autonomous networks (AN) support identification of parameters which can be optimized, including ML parameters, based on the use case.

NOTE – Example of use case is parameter optimization for massive MIMO large-scale antenna array, including horizontal lobe width, vertical lobe width, azimuth, dip angle and beam number.

- AN-UC08-REQ-002: It is critical that autonomous networks (AN) support identification of data which can be collected to analyse and infer, based on the use case.

NOTE – Examples of data are distribution of UE, the target RSRP/SINR distribution in the current scenario.

- AN-UC08-REQ-003: It is critical that autonomous networks (AN) support identification of KPIs which need to be optimized.

NOTE – Examples of KPIs are system capacity and QoE.

- AN-UC08-REQ-004: It is critical that autonomous networks (AN) support optimization of KPIs in distributed deployments which require multicell coordination.

7.8.2 Use case specific figures

None.

7.9 Network resource allocation for emergency management based on closed loop analysis

<i>Use case id</i>	FG-AN-usecase-9
<i>Use case name</i>	Network resource allocation for emergency management based on closed loop analysis
<i>Base contribution</i>	FGAN-I-090-R2
<i>Creation date</i>	22/April/2021
<i>Use case context</i>	Discussions during [FGAN-I-055-R1], [FGAN-I-054-R1], [FGAN-I-072]
<i>Use case description</i>	<p>Telecommunication systems are critical pillar of emergency management. A set of hierarchical AI/ML based closed loops could be used to intelligently deploy and manage slice for emergency responders in the affected area. A higher closed loop in the OSS can be used for detecting which area is affected by the emergency and deploy a slice for emergency responders to that area. It can then set a resource arbitration policy for the lower closed loop in RAN. The lower loop can use this policy to intelligently share RAN resources between the public and emergency responder slice. It can also intelligently manage ML pipelines across the edge and emergency responder devices by using split AI/ML models or offloading of inference tasks from the devices to the edge.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. MNO may instruct OSS to detect certain set of emergencies and provide connectivity to emergency responders according to predefined SLA. <p>NOTE – e.g., this input may be provided using an operator intent.</p>

	<p>2. OSS might deploy a closed loop to achieve this. It might collect data from sources like network analytics data, social media scraping, input from emergency responders etc. NOTE – e.g., such inputs may be provided from nRT-RICs or other xNFs in the network.</p> <p>3. OSS might use AI/ML models to detect emergency and deploy an ER slice to the location. It might also create high level strategy/policy to reallocate resources among the slices. NOTE 1 – e.g., such closed loops may be hosted in non-RT RIC and may be used for predictive resource allocations to specific edge locations based on predicted needs, in turn based on detected emergency. NOTE 2 – The policy to reallocate resources may depend, among other things, on the type of emergency e.g., a natural disaster, earthquake, a law and order situation, traffic accidents, etc.</p> <p>4. RAN domain might use this high-level strategy/policy and possibly other inputs from emergency responders to create a closed loop to arbitrate resources among RAN NSSIs. NOTE – e.g., such closed loops may be hosted nearer to edge e.g., nRT RIC. The policy input from higher loop may indicate, among other things, the different sources of data for the lower loop.</p> <p>5. RAN domain closed loop might also decide to offload inference tasks from ER devices to the edge or use split AI/ML model to run inference tasks on edge and ER device. This decision might be taken based on available network and compute resources. NOTE – e.g., some layers of the AI/ML model may be hosted in the wearable devices of the emergency responders, which will help in say locating of persons under distress using various inputs.</p> <p>Relation with autonomous behavior</p> <p>1. Workflows for the closed loops are independent of each other. The only interaction between closed loops is via high level intents over the inter-loop interface.</p> <p>2. Closed loops can create new closed loops in other network domains without human intervention.</p> <p>3. Although loops are deployed in hierarchical fashion, each loop has the ability to evolve independently. It can use different models and ML pipelines as required. Each loop may move up or down the autonomy levels as defined in [ITU-T Y.3173].</p> <p>4. Closed loops have ability to split and provision AI/ML models to other closed loops in automated fashion.</p> <p>5. By making closed loops in edge domain autonomous, we also enable lesser orchestration delay, better privacy and flexibility for verticals (e.g., industrial campus networks).</p> <p>6. Higher loops can use historical knowledge available to them to optimize and generalize lower loops using high-level intent. This increases efficiency of lower loops while preserving their autonomy. (e.g., higher loop might know certain kind of ML models are good for cyclone emergency management based on previous cyclones.)</p> <p>NOTE – This use case might be well aligned with the use case "Composable, hierarchical closed loops" in [FGAN-I-072] and others above.</p>
<p><i>Open issues (as seen by the proponent)</i></p>	<p>1. The "propagation" and "escalation" of intents is something which needs study.</p> <p>2. Mixing the concept of "declarative" policies with "top-level" intent with the concept of (propagation/escalation) needs study.</p> <p>3. Recursive decomposition coupled with recursive policy mapping.</p>

	<p>4. The components ("nodes") of the high-level service are decomposed into more concrete services (possibly recursively). Declarative policies must be "translated" into more concrete declarative policies on the decomposed services in conjunction. – how to do this?</p> <p>5. By coupling "event/condition/action" control loops with TOSCA's substitution mapping feature, you can make these control loops "cascading", i.e., they can propagate down from high-level abstract "intent" statements to low-level device reconfigurations, and they can escalate back up if necessary. This is an aspect of TOSCA policies that needs study.</p>
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behavior itself.
<i>Notes on priority of the use case</i>	High – Has the potential to impact future network architectures and use cases.
<i>Reference</i>	[b-ETSI GS ZSM 001]

7.9.1 Use case requirements

Critical requirements

- AN-UC09-REQ-001: It is critical that AN allow interaction between closed loops via high level intents.

NOTE – Closed loops may create new closed loops in other network domains without human intervention.

- AN-UC09-REQ-002: It is critical that AN allow each loop to evolve independently, using different analytical, optimization mechanisms including ML models and ML pipelines as required.

NOTE – Each loop may move up or down the autonomy levels as defined in [ITU-T Y.3173].

Expected requirements

- AN-UC09-REQ-003: It is expected that Closed loops have ability to provision or recommend AI/ML models to other closed loops in automated fashion.
- AN-UC09-REQ-004: It is expected that closed loops in edge domain may be autonomous, in order to enable lesser orchestration delay, better privacy and flexibility for verticals (e.g., industrial campus networks).
- AN-UC09-REQ-005: It is expected that higher loops use the knowledge base available to them to optimize and generalize lower loops using high-level intent.

NOTE – This increases efficiency of lower loops while preserving their autonomy. (e.g., higher loop might know certain kind of ML models are good for cyclone emergency management based on previous cyclones.)

7.9.2 Use case specific figures

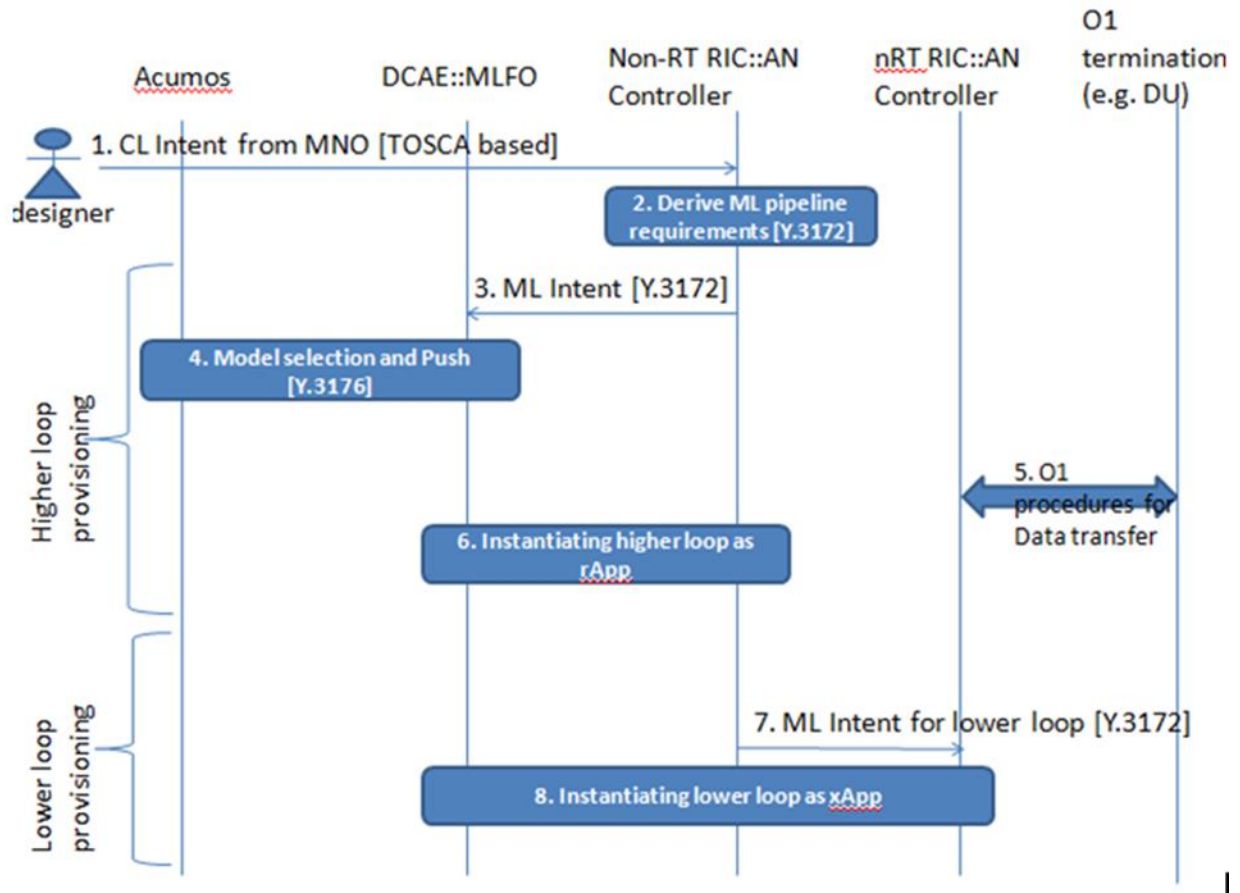


Figure 10 – Actor interaction for Network resource allocation for emergency management based on closed loop analysis

NOTE 1 – Create a high-level abstract model for closed loops, and then create declarative policies for that high-level model that express the "intent" of creating ML pipelines. The components ("nodes") of the high-level service are decomposed into more concrete services (possibly recursively). Declarative policies must be "translated" into more concrete declarative policies on the decomposed services in conjunction. For example, "non-RT" level service may impose certain closed loop requirements on a RIC that implements the ML pipeline. "nRT" level service may impose some other closed loop requirements on a RIC that implements that ML pipeline. This recursive decomposition coupled with recursive policy mapping happens all the way down until service components can get realized on the available resources. At that point, the low-level declarative policies must be translated somehow into imperative policies (e.g., if jitter exceeds a certain threshold, re-prioritize the traffic associated with the service).

NOTE 2 – "Imperative" policies that use the "event/condition/action" pattern, vs. declarative policies use a "capabilities/context/constraints" pattern. Declarative policies are more suitable for top-level "intent" statements, but they need to be translated (by the orchestrator) into corresponding "imperative" policies in order to be actionable. The "propagation" and "escalation" of intents: the "event/condition/action" statements are the control loops you're referring to that make sure that service components comply with desired behavior at all times. By coupling "event/condition/action" control loops with TOSCA's substitution mapping feature, you can make these control loops "cascading", i.e., they can propagate down from high-level abstract "intent" statements to low-level device reconfigurations, and they can escalate back up if necessary.

NOTE 3 – The events are generated (using notifications) by nodes in the service topology model. The conditions are evaluated based on attribute values of nodes in the service topology model. The actions are performed on the service topology model first, and then propagated to the external world (the "resources").

7.10 Inter-domain service automation (IDSA) – for microfinance

<i>Use case id</i>	FG-AN-usecase-010
<i>Use case name</i>	Inter-domain service automation (IDSA) – for microfinance
<i>Base contribution</i>	AN-I-060
<i>Creation date</i>	29/March/2021
<i>Use case context</i>	Discussions regarding cloud interoperability in 5G use case lab
<i>Use case description</i>	<p>Microfinance applications may be hosted by non-experts in 5G or any form of cloud / ICT technologies. The end-user requirements are domain-specific e.g., loan management, banking account/ledger management etc. The main stakeholders who are enterprises (e.g., banks) may be knowledgeable and would like to focus only in their business workflows (as against cloud / ICT technologies). The underlying cloud infrastructure (for that matter the application design) and the network infrastructure (5G, 6G, or (x+1)G) is immaterial to a bank / finance manager. However, from a technology perspective the following rewards are desired to be reaped:</p> <ol style="list-style-type: none"> 1. Give the best end-user experience: e.g., reduce down-time for services, reduced latencies for services, security and data privacy, intelligent services, by exploiting the best cloud service deployment for the microfinance application. e.g., edge, load balancing, secure messaging across multi-cloud, hybrid-cloud, AI/ML services via distributed cloud, etc. 2. Insulate the end-user from xG -> (x+1)G migration: by providing interoperable, standard, backward compatible networking abstraction technologies. Integrating service lifecycle management pipeline provides agility to service development and testing. 3. Mitigate the risk of increased integration service costs: by using open source technologies, standards, benchmarking, automating in test beds. 4. Automate: reducing human involvement reduces training costs for banks, operational costs in networks and brings other benefits like intelligent fault isolation without depending on 3rd party service providers. <p>NOTE – Devops [b-ISO/IEC 23167], [b-ITU-T Y.3515], CI/CD [b-ITU-T Y.3525] are examples of service lifecycle management pipelines.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Intent based cloud service specification. 2. Processing of intent and development, validation in Sandbox/testbed. Testbed components (e.g., simulators, data models) are selected based on intent. 3. Evaluation and analysis of test results based on key parameter indices (KPI) specifications in the intent. 4. Derivation of optimal configuration, cloud service deployment, management and orchestration. 5. Intent based network service deployment, management and orchestration. 6. Single "cockpit" for monitoring the services 7. Autonomous, Intelligence-guided, technology-agnostic migration of services from one version of underlying technology to another e.g., xG -> (x+1)G and edge -> fog. 8. Reports from various parts of the underlying technologies are provided to humans in regular intervals or event based.
<i>Open issues (as seen by the proponent)</i>	1. Handling of accounting for such services is not clear.
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.

<i>Notes on priority of the use case</i>	High – Enables vertical driven applications and network evolution.
<i>Reference</i>	

7.10.1 Use case requirements

Critical requirements

- AN-UC10-REQ-001: It is critical that autonomous networks (AN) consider inputs from industry vertical solution provider regarding the required service characteristics, using an intent-based mechanism, while deciding the development and deployment options for industry vertical applications and network services.

NOTE – AN can autonomously decide the best possible development and deployment option for network services which can support the verticals. This has to be based on the requirements of the applications [ITU-T Y.3178]. E.g., for banking applications, service characteristics may include latency on banking transactions, mean time between service failures, level of privacy of each field in the customer profile, etc. Examples of deployment options may include edge, core cloud, enterprise network, using specific hardware etc.

- AN-UC10-REQ-002: It is critical that, AN abstracts the management (creation, deletion and update) of the industry vertical applications and network services, from the industry vertical solution provider.

NOTE – Underlying domain orchestration, network specific technologies and APIs are abstracted by AN towards the industry vertical solution provider. E.g., banking applications may be hosted as web applications (on popular web frameworks with or without an accompanying mobile component), enterprise applications (e.g., J2EE based). They may be instantiated as cloud-native applications, may use distributed architecture across private/public clouds etc. Service management infrastructure supporting the applications may include brokers, workflow managers and schedulers. Irrespective of such deployment and management variance, AN provides abstracted interfaces to verticals which hides such complexities.

- AN-UC10-REQ-003: It is critical that, AN validates any changes to the application and network services in a sandbox environment before applying it in the network.

NOTE – Autonomous behaviour may result in automated creation, deletion and update of applications and/or network services. The impact of such modified applications and/or network services has to be studied before they are applied in the network. This may be done by using a testbed or sandbox with simulators or even a digital twin-based environment. Specific emphasis may be applied on maintaining compatibility of the modifications with the applications and network services in the network.

- AN-UC10-REQ-004: It is critical that, AN continuously monitors the application and network services in the network.

NOTE – Monitoring may be done to find erroneous behaviour, faults, gaps in architecture, design, bugs, etc. Monitoring may identify the gaps in end-to-end service implementations with respect to changing intents of the verticals. Thus, monitoring may also be used to find the need for evolution in underlying network domains.

- AN-UC10-REQ-005: It is critical that, AN produces regular and asynchronous reports for human consumption.

NOTE – Reports to humans may summarize all monitored values, analysis, decision points and explanations for such decisions by the AN.

Expected requirements

- AN-UC10-REQ-006: It is expected that, AN provide automated triggers to service lifecycle management pipeline for management (creation, deletion and update) of application and network services.

NOTE 1 – Devops [b-ISO/IEC TS 23167], [b-ITU-T Y.3515], CI/CD [b-ITU-T Y.3525] are examples of service lifecycle management pipelines.

NOTE 2 – As part of management of applications, AN may analyse the gaps, faults and issues in the current design and implementation of end-to-end service. Mitigation of such issues may be triggered to the devops pipeline. However, the level of automation of the solution may depend on the capabilities of the devops pipeline.

- AN-UC10-REQ-007: It is expected that, the AN configuration includes the set of reference points which may be used for integration into end-to-end network services and applications.

NOTE – Even though AN exposes abstracted interfaces for application management to verticals, to achieve the integration of network services, AN may use open interfaces or closed black boxes. The availability of open interfaces and corresponding components is to be made known to AN via configurations. Such configurations may be dynamically changing based on availability of new components and interfaces.

- AN-UC10-REQ-008: It is expected that, the AN proposes "recipes" of network services and applications which may satisfy a particular intent from the vertical.

NOTE – Recipes may include a combination of existing application components, network service components, corresponding configuration options, etc.

- AN-UC10-REQ-009: It is expected that, the AN is updated at runtime by the underlying domain orchestration about the supported set of reference points in the domain, available set of network service and application components, which may be used for integration into end-to-end network services and applications.

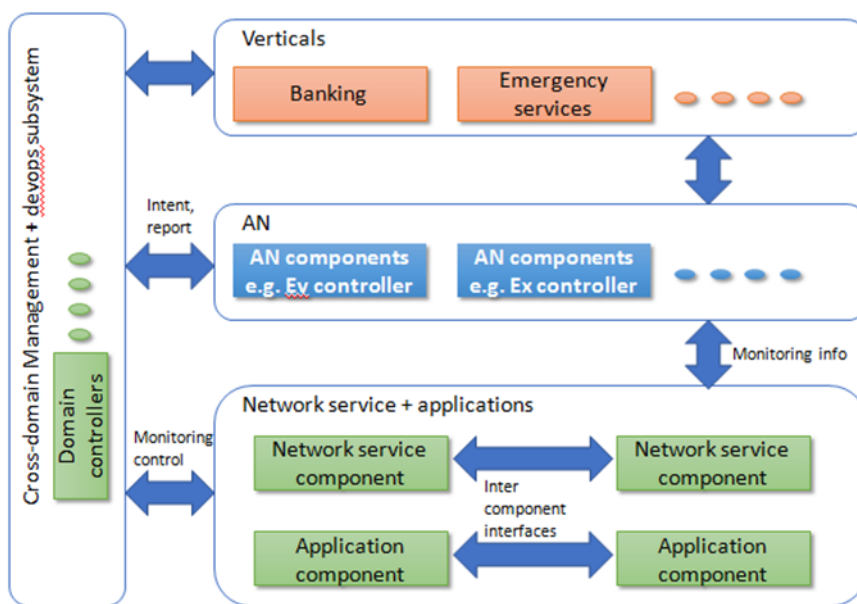
NOTE – Runtime changes, triggered by the operator or 3rd parties, in the underlying domains are made aware to the AN. Such updates may be abstracted and passed by the AN to the verticals, where relevant, for information or policy decisions.

Added value requirements

- AN-UC10-REQ-010: It is of added value that, the AN proposes a modified "recipe" of network services and applications which may bridge a gap, fix a fault or solve issues in the current design and implementation of end-to-end services.

NOTE – Modified recipe may be based on analysis of gaps, issues, or faults encountered while monitoring of network services and applications.

7.10.2 Use case specific figures



Editor note:
 1. For now, devops and management are combined.
 2. Number the arrows and call out the functionality of each.

Figure 11 – Inter-domain service automation

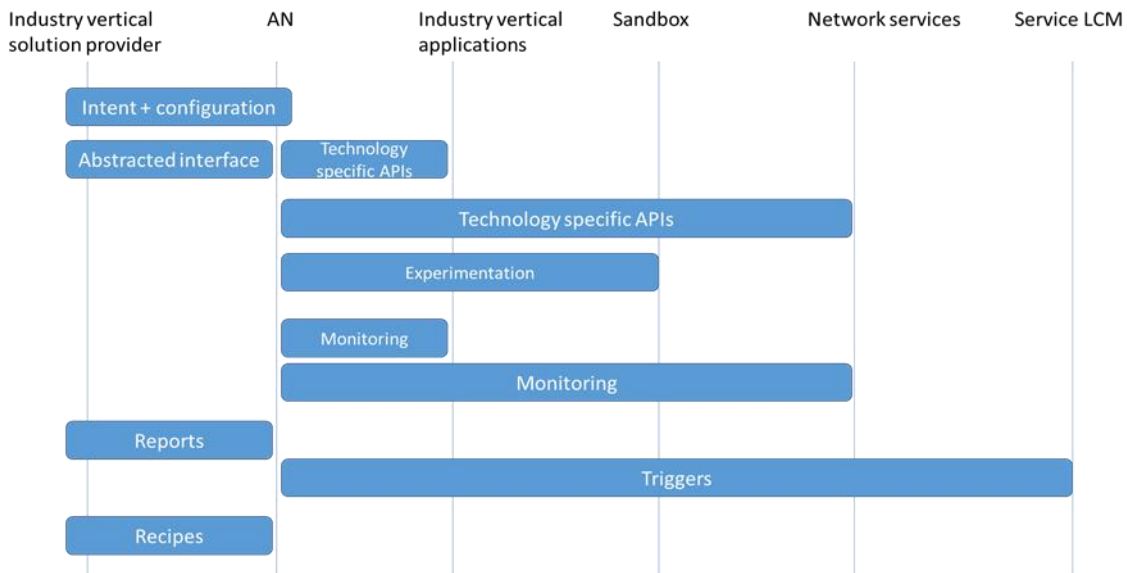


Figure 12 – Actor interaction for Inter-domain service automation (IDSA) – for microfinance

7.11 Autonomous vertical-driven edge service and middle-mile connectivity for rural financial inclusion (FI)

<i>Use case id</i>	FG-AN-usecase-011
<i>Use case name</i>	Autonomous vertical-driven edge service and middle-mile connectivity for rural financial inclusion (FI)
<i>Base contribution</i>	AN-I-060
<i>Creation date</i>	31/March/2021
<i>Use case context</i>	Discussions regarding rural broadband architecture(s) and feedback from survey to banks during summer 2020
<i>Use case description</i>	<p>Digital Financial Inclusion in many geographies are limited because of lack of network availability and low reliability of connection. Frequently bank branches have to fall back on costly and complex connectivity through satellites. Other than the capital and operational aspects, the bank staff also needs to handle the link failover and maintenance activities in case of issues. Furthermore, such solutions do not allow local community to utilize the network.</p> <p>Autonomous last hop connectivity both through 4G/5G as well as other non 3GPP heterogeneous networks need to be seamlessly enabled to operate in an affordable manner. The solution is likely to provide the following benefits:</p> <ol style="list-style-type: none"> 1. Based on the requirements from the verticals, provide connectivity for rural FI sites independent of network providers/large-Telco. 2. Allow the connectivity to be shared with local community to ensure better return on investments. 3. Edge compute and related infrastructure can enable more compelling deployments for digital FI as well as other verticals (e.g., short term tele-commute / interviews etc.). 4. Automate operations and security audits: reducing human involvement in maintaining and running the edge / last hop reduces training costs for banks, operational costs in networks and brings other benefits like intelligent fault isolation without depending on 3rd party service providers, continuous audit of deployed solution for security etc. <p>Following are related sub-systems and associated steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Deploy micro-servers/nano-data-centres for the edge. 2. Enable heterogeneous network connectivity.

	<ol style="list-style-type: none"> 3. Autonomous, Intelligence-guided handling of alignment / interference / mobility related challenges for various last hop approaches. 4. Automate on-boarding of community users and community specific apps, their billing/payments etc. 5. Single "cockpit" for monitoring the services and health of infra to local-bank-staff/managed-service-provider. 6. Reports from various parts of the underlying technologies are provided to humans in regular intervals or event based. This includes sharing of usage details with authorized management systems. These reports may be used for tracking the usage at a granular level, mapped to the vertical and the tracking the corresponding benefits from the infrastructure e.g., for the purpose of extending subsidies to such infrastructure.
<i>Open issues (as seen by the proponent)</i>	1. Policy framework for such an autonomous community network anchored by a specific vertical needs to be understood.
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	High – Enables vertical driven applications and network evolution.
<i>Reference</i>	

7.11.1 Use case requirements

Critical requirements

- AN-UC11-REQ-001: It is critical that autonomous networks (AN) utilize heterogeneous network connectivity options at the edge, in the last-mile, including the commissioning, provisioning, configuration, integration, maintenance and optimization, in a seamless, real-time and easy-to-use manner.

NOTE – Especially in rural settings the technology of choice may vary considerably depending on various factors like availability of technology, ease of deployment, low power consumption, etc. Currently, in such deployments, it is invariably upon the industry vertical solution provider to also take on the responsibility of integrating and maintaining these varied last mile connectivity options. It is important that AN brings together various such technologies under one umbrella, at the edge, to provide seamless integration and maintenance. Some of the operations in the lifecycle of the last mile connectivity may need real-time interventions, some of it may need deep domain expertise and some of it may require training – all of which may not be possible in certain rural settings.

- AN-UC11-REQ-002: It is critical that autonomous networks (AN) enable sharing of the various network connectivity options at the edge, across the community driven industry vertical applications.

NOTE – With an emphasis on providing maximum connectivity and application services to the local community, the best available option for connectivity needs to be chosen, if needed, dynamically.

- AN-UC11-REQ-003: It is critical that autonomous networks (AN) enable onboarding of industry vertical applications at run-time.

NOTE – Evolution of needs in a local community may result in changing application requirements.

- AN-UC11-REQ-004: It is critical that autonomous networks (AN) enable common, open, interoperable, adaptable mechanisms for managing end-users, based on the needs of the local community.

NOTE – Onboarding, billing, problem-resolutions and other end-user management functions need to be agnostic, community-driven at the edge. Based on the use cases, the mechanisms for end-user management have to adapt. E.g., for low-mobility rural areas, a relevant tariff plan needs to be offered.

- AN-UC11-REQ-005: It is critical that autonomous networks (AN) enable a single-window of monitoring the heterogeneous underlying technologies.

NOTE – Complexities of monitoring, administering, maintaining the complexities of the underlying technologies need to be hidden from the industry vertical solution provider as well as the local communities.

7.11.2 Use case specific figures

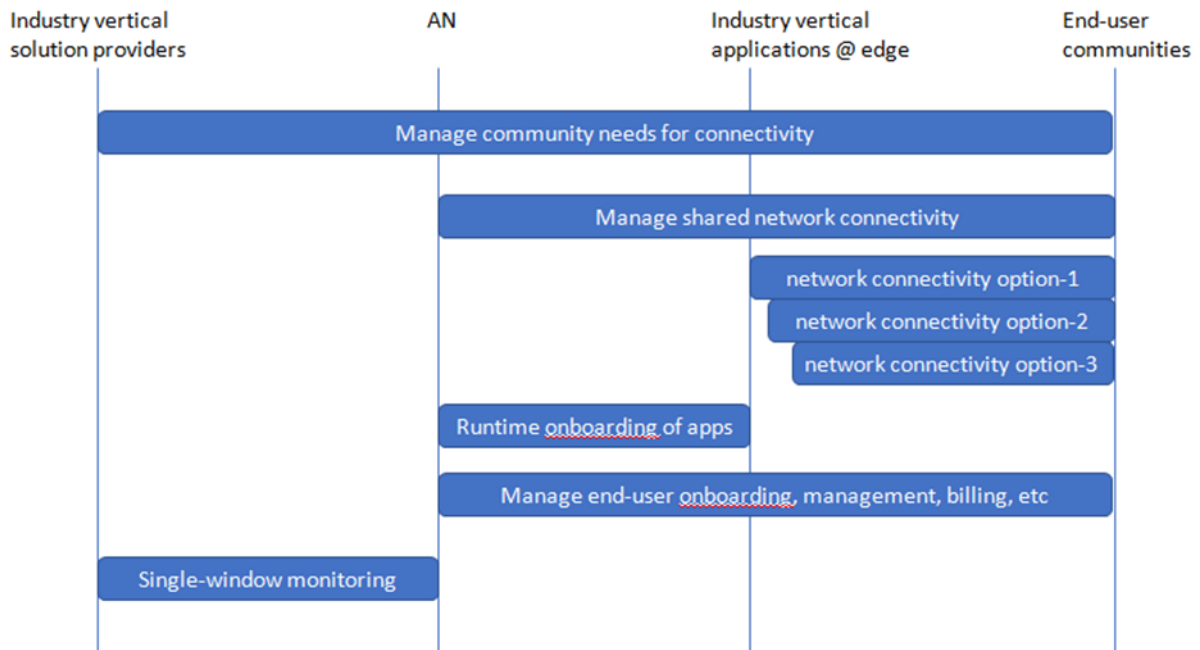
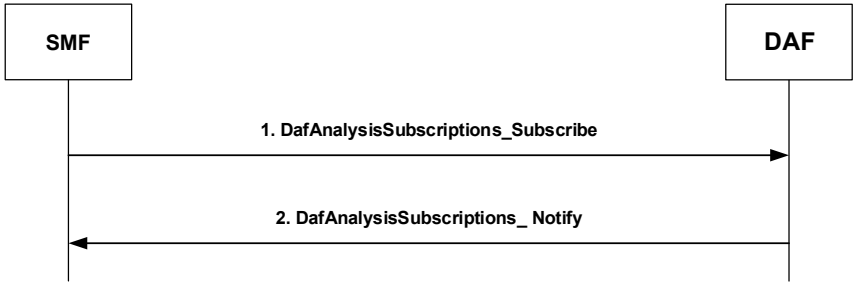


Figure 13 – Actors in Autonomous vertical-driven edge service

7.12 Signalling flows for autonomous IMT-2020 network

<i>Use case id</i>	FG-AN-usecase-012
<i>Use case name</i>	Signalling flows for autonomous IMT-2020 network
<i>Base contribution</i>	AN-I-064
<i>Creation date</i>	21 March 2021
<i>Use case context</i>	Signalling flows between data analysis function and other network functions, to achieve self-analysis and self-optimization
<i>Use case description</i>	<p>Data analysis function (DAF) is introduced in IMT-2020 network [b-ITU-T Y.3104]. The signalling flow between DAF and other network functions (e.g., SMF, PCF, NACF and AF) describes data collection and analysis result providing.</p> <p>The procedure in Figure 1 is used by DAF to collect data on event (s) related to SMF by invoking SmfEventSubscription service.</p> <div style="text-align: center;"> <pre> sequenceDiagram participant DAF participant SMF Note over DAF, SMF: 1. SmfEventSubscription_Subscribe/ SmfEventSubscription_Unsubscribe DAF->>SMF: Note over SMF, DAF: 2. SmfEventSubscription_Notify SMF-->>DAF: </pre> </div> <p>Figure 1 Signalling flow for Data Collection from SMF</p> <ol style="list-style-type: none"> The DAF subscribe to or unsubscribe from a (set of) Events (e.g., UE IP address, UP path change, PDU Session Establishment/ Release, and etc.) by invoking the SmfEventSubscription_Subscribe service operation.

	<p>2. The SMF notifies the DAF (e.g., with the event report) by invoking SmfEventSubscription_Notify service operation.</p> <p>The procedure in Figure 2 is used by NF service consumers (e.g., SMF) to request analytics information from DAF by invoking DafAnalysis_Request service.</p> <div style="text-align: center;">  <pre> sequenceDiagram participant SMF participant DAF SMF->>DAF: 1. DafAnalysisSubscriptions_Subscribe DAF-->>SMF: 2. DafAnalysisSubscriptions_Notify </pre> </div> <p>Figure 2 Signalling flow for Analytics Subscribe/Unsubscribe from DAF</p> <ol style="list-style-type: none"> SMF subscribes to or unsubscribe from a (set of) data analytic events by invoking the DafAnalysisSubscriptions_Subscribe service operation. Subscription requirements of data analytic events may include: <ul style="list-style-type: none"> – Load information of UPF The DAF notifies the SMF about analysis events by invoking DafAnalysisSubscriptions_Notify service operation.
<i>Use case category</i>	Cat 1: Autonomous behaviour in IMT-2020 network.
<i>Reference</i>	[b-Q.IMT2020-PIAS]

7.12.1 Use case requirements

Critical requirements

- AN-UC012-REQ-001: It is critical that autonomous networks (AN) enable flexible provisioning and subscription of analysis parameters in network functions.

NOTE – Examples of analysis parameters are events, notifications, corresponding information and event handing controllers. Network functions may dynamically provision or subscribe to controllers and corresponding parameters.

7.12.2 Use case specific figures

None.

7.13 Plug/play of network instance

<i>Use case id</i>	FG-AN-usecase-013
<i>Use case name</i>	Plug/play of network instance
<i>Base contribution</i>	AN-I-072
<i>Creation date</i>	13 April 2021
<i>Use case context</i>	Discussions regarding FGAN-I-017 and deep dive
<i>Use case description</i>	<p>Benefits of open architecture approach include:</p> <ul style="list-style-type: none"> *) Reducing CAPEX through a prosperous multi-vendor ecosystem with scale economics. However, more the number of interfaces, more the effort in integration. This needs to be mitigated using automation. *) Rich application space enabled using hierarchical controllers. The hierarchical control loops with varying time criticalities (<10 ms (at edge) < 1 s (at near edge) < multi-second (at orchestrator)) were discussed in FGAN-I-017. However, provisioning of applications at various levels and corresponding coordination with capabilities of the network functions is a challenge.

	<p>In this context, we introduce the use case "Plug/play of network instance" in open architecture.</p> <p>NOTE – The network instance can be network resource, network function, network slice and network services [b-ETSI GS ZSM 001].</p> <p>Following are related steps in this use case scenario:</p> <p>9. Addition of SRCs [ITU-T Y.3172]: network instance is plugged into the network. Data collection functions supported by this new SRCs are analysed.</p> <p>10. Bottom-up bootstrapping of infrastructure layer (using cloud orchestration), network as a service (NaaS, using ONAP), services layer (using service orchestration), based on these new SRCs.</p> <p>OR</p> <p>Top down bootstrapping of apps, services, NaaS, infrastructure, based on these new SRCs.</p>
<i>Open issues</i>	E2E automation frameworks for composition of infrastructure, NaaS and services do not exist.
<i>Use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Reference</i>	

7.13.1 Use case requirements

Critical requirements

- AN-UC013-REQ-001: It is critical that autonomous networks (AN) enable plug and play of network functions in the underlay and subsequent seamless participation of such network functions in the AN functions.

NOTE – Examples of AN functions are creation and hosting of controllers. Plug and play may be executed by manual or autonomous mechanisms.

7.13.2 Use case specific figures

None.

7.14 "Generative adversarial Sandbox": (or hybrid closed loops)

<i>Use case id</i>	FG-AN-usecase-014
<i>Use case name</i>	"Generative adversarial Sandbox": (or hybrid closed loops)
<i>Base contribution</i>	FGAN-I-017
<i>Creation date</i>	13/Apr/2021
<i>Use case context</i>	Discussions regarding FGAN-I-017 and deep dive
<i>Use case description</i>	<p>In addition to open interfaces between various RAN components, a rich ecosystem of simulators is evolving. This allows implementation of various "hybrid" closed loops – part of the closed loop (e.g., data generation) is implemented in simulators whereas rest of the closed loop (e.g., analysis and action) are implemented in another part of the test network using real network functions (NF).</p> <p>In this context, we introduce the use case "Generative adversarial Sandbox: or hybrid closed loops".</p> <p>Following are related steps in this use case scenario:</p> <p>5. Based on the inputs from the NF (e.g., data from SRC) and existing closed loops, simulator configurations and capabilities are autonomously scripted.</p> <p>6. Hybrid closed loops are autonomously composed – with parts of the closed loop in real NF and parts of it in simulators.</p> <p>7. Similar to Generative adversarial Networks, hybrid closed loops are evaluated and tested using 2-part network – one simulated and another real network functions.</p>

	8. The results are analysed and ranked.
<i>Open issues</i>	4. Does the open interfaces extend to simulators? 5. How to rank the experiments? 6. Data models may be different in various interfaces with simulators.
<i>Use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Reference</i>	

7.14.1 Use case requirements

Critical requirements

- AN-UC014-REQ-001: It is critical that autonomous networks (AN) enable creation of hybrid closed loops with parts of the closed loops hosted in real network functions as against other parts of it in simulated network functions.

NOTE – Examples of parts of closed loops are modules which generate data, modules which implement domain specific functions, modules which provide APIs for implementation of adapting decisions from controllers.

- AN-UC014-REQ-002: It is critical that autonomous networks (AN) enable testing and validation of closed loops using the parts of hosted in simulated network functions.

NOTE – Examples of such testing are robustness related test scenarios, security and vulnerability testing scenarios.

7.14.2 Use case specific figures

None.

7.15 Open, integrated, log analysis

<i>Use case id</i>	FG-AN-usecase-015
<i>Use case name</i>	Open, integrated, log analysis
<i>Base Contribution</i>	FG AN 2nd virtual meeting
<i>Creation date</i>	FGAN-I-017
<i>Use case context</i>	Discussions regarding FGAN-I-017 and deep dive
<i>Use case description</i>	Fault prediction and isolation based on log analysis is an important existing use case. Logs are generally implemented in unstructured text with no standard formats. With a disaggregated network service implementation, correlating logs from various vendors becomes a challenge. This throws cold water on the fault prediction and fault isolation algorithms based on logs. In this context, we introduce the use case "Open, integrated, log analysis". Following are related steps in this use case scenario: 4. Collection of logs from various open interfaces and NFs 5. Correlation and Analysis of the collected logs, across various open interfaces and NFs. 6. Identification of optimization mechanisms based on log analysis.
<i>Open issues</i>	
<i>Use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Reference</i>	

7.15.1 Use case requirements

Critical requirements

- AN-UC015-REQ-001: It is critical that autonomous networks (AN) enable correlation and identification of relevant logs, their access using open interfaces, analysis and resulting optimization of network underlays to apply specific adaptations.

7.15.2 Use case specific figures

None.

7.16 Compose-able, hierarchical closed loops

<i>Use case id</i>	FG-AN-usecase-016
<i>Use case name</i>	Compose-able, hierarchical closed loops
<i>Base contribution</i>	FGAN-I-017
<i>Creation date</i>	13/Apr/2021
<i>Use case context</i>	Discussions regarding FGAN-I-017 and deep dive
<i>Use case description</i>	<p>There are different automation loops in different levels of the architecture. High level use cases (like log-analysis based fault prediction) require access to capabilities of various network instances. This in turn may be provided by multiple vendors or opensource providers. Thus, provisioning and management of closed loops should be driven hierarchically.</p> <p>In this context, we introduce the use case "Compose-able, hierarchical closed loops".</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none">5. Declarative specifications decide the high level aspects of closed loops.6. They are in turn correlated with declarative specifications for network services.7. These are in turn used to generate detailed declarative specifications for closed loops in different parts of the network.8. Orchestrators at various levels generate commands to provision and manage the closed loops based on these generated declarative specifications.9. The declarative specifications and/or closed loop components may be stored/updated for regeneration of closed loops at any point of time.
<i>Open issues</i>	
<i>Use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Reference</i>	

7.16.1 Use case requirements

Critical requirements

- AN-UC016-REQ-001: It is critical that autonomous networks (AN) enable composition of hierarchical closed loops using declarative specifications.
- AN-UC016-REQ-002: It is critical that autonomous networks (AN) enable derivation of controllers or closed loops at various levels of the network.
- AN-UC016-REQ-003: It is critical that autonomous networks (AN) enable management of declarative specifications of closed loops or controllers.

NOTE – Management operations on declarative specifications may include creation, storage, update, delete, etc.

7.16.2 Use case specific figures

7.17 Quality of Experience (QoE) Prediction as-a-Service (QPaaS)

<i>Use case id</i>	FG-AN-usecase-017
<i>Use case name</i>	Quality of Experience (QoE) Prediction as-a-Service (QPaaS)
<i>Base contribution</i>	FGAN-I-110
<i>Creation date</i>	16 June 2021
<i>Use case context</i>	"Review of academic papers on QoE Predictions." FGAN-I-110
<i>Use case description</i>	<p>Intelligent and autonomous troubleshooting is a crucial enabler for the current 5G and future 6G networks. Autonomous troubleshooting is challenging for several reasons, one of which is the availability of a wide range of applications that future networks will support.</p> <p>Traditionally, the methods to gain insight into the delivered quality of service and the users' experience have been through controlled laboratory experiments, where users' opinions have been collected. The results are then reported in Mean Opinion Scores (MOS), corresponding to the average of users' views. These methods are often referred to as subjective quality assessment, and there are standardized methods for conducting them.</p> <p>In this use case, an application or network service (NS) provider will use a QoE-Prediction-as-a-Service (QPaaS) autonomous system to conduct and follow-up QoE measurement and prediction.</p> <p>Firstly, the autonomous system will conduct subjective tests to measure the user experience from participating users. The locations and specifications of which users will be selected and how the users' responses affect the QoE will depend on the application and will be learned by the autonomous system. The autonomous system will also measure relevant user parameters to map user opinions and application KPIs.</p> <p>Secondly, the autonomous system will follow applicable network KPIs and map network and application KPIs.</p> <p>Thirdly, based on this mapping, the autonomous system will enable the application provider to predict the QoE of its users based on network KPIs regardless of their participation. The autonomous system continuously (or periodically) improves the prediction accuracy by random subjective tests or user behavior analysis.</p> <p>Related steps in this use case scenario are:</p> <ul style="list-style-type: none"> • Application or network service (NS) provider demands and deploys QoE prediction as-a-service (QPaaS) from a third-party server. <ul style="list-style-type: none"> – Application or NS provider provides a mechanism to collect/use user feedback and network metrics. • Identify a method of measurement for QoE: <ul style="list-style-type: none"> – Perform subjective tests, e.g., video streaming, two-way communications, etc. User opinions on a scale of 1-5 or thumbs up/down. – Perform user behavior analysis, e.g., gaming, AR/VR, driver assistance, etc. In a group of gamers connected via various CSP (communication service providers), if the gamers from a particular CSP face delays or a specific cell site (geographic area) is facing latency, the gaming scores and avatar-behaviour itself leave enough clues on the QoE. Similarly, on AR/VR, the level of engagement/interaction, or in assisted driving, the level of coordination between vehicles, can be measured. • QPaaS server, collects/processes network and application KPIs. • QPaaS server determines a mapping between application KPIs and application QoE metric (MoS) using (supervised) machine learning.

	<ul style="list-style-type: none"> – This mapping may be used by the application server for future objective testing of user QoE. • QPaaS server collects/processes relevant network service KPIs and forms a mapping between network KPI and application KPI (or MoS) using (supervised) machine learning. <ul style="list-style-type: none"> – The NS provider may use this mapping for future objective testing of network performance given the application. • Perform periodic verification with subjective tests/user feedback, and improve learning based on the results. <p>Due to various applications, QoE measurement and prediction is a significant issue in future networks. The network should be able to autonomously perform QoE measurement and mapping of network KPI to QoE metrics.</p> <p>NOTE – As applications and network services evolve, so do their corresponding KPIs and the mappings (user satisfaction parameters, to application KPIs, and to network KPIs). See related open issues below which handles information exchange between evolving applications and NS and QPaaS.</p>
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> • How are application KPIs provisioned in the QPaaS server? How is this done for new applications? Can we transfer knowledge gained from Sandbox in onboarding new applications and services and corresponding mappings? • How do you access relevant network KPIs? • How are network KPIs provisioned in QPaaS server? How is this done for new NS (network services)? • How do you select users to collect data from? • How do you ensure the privacy of the user and network service data? E.g., federated learning may allow privacy for user data. • How much training data do you need to collect for objective QoE measurement? • Can you perform reinforcement learning to improve the QoE model?
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to the application of autonomous behavior in the network.
<i>Notes on priority of the use case</i>	High. QoE is an essential metric for user satisfaction. Although QoE measurement is well analysed in the context of video streaming, there is no general definition for a wide variety of applications. This autonomous service will allow future application developers to model and track the QoE of their applications online. QoE tracking will enable application developers to update network slices dynamically.
<i>Reference</i>	[b-Jahromi], [b-Pierucci], [b-Bouraquia], [b-Liu]

7.17.1 Use case requirements

Critical requirements

- AN-UC017-REQ-001: It is critical that autonomous networks (AN) use both subjective information from users and QoE information derived and analysed from network services to arrive at the application QoE metric.

NOTE – Subjective information from users may include user opinions and subject measures e.g., opinions on a scale of 1-5 or thumbs up/down about a streamed video. Examples of QoE information derived and analysed from network services are the level of engagement/interaction in an online game, or analysis of gaming scores and avatar behaviour.

- AN-UC017-REQ-002: It is critical that autonomous networks (AN) learn and update the process of information collection from users and derivation from network services.

NOTE 1 – For example the parameters collected, the mechanisms for collecting and the sample set for collection may be learnt and updated. Also the mapping between the application QoE metric and the information collected from users and derived from network services may evolve over a period of analysis.

NOTE 2 – The mapping between the application QoE metric and the information collected from users and derived from network services may be modelled using AI/ML techniques.

- AN-UC017-REQ-003: It is critical that autonomous networks (AN) evolve and update the mapping between application QoE metric, network KPIs and application KPIs.

NOTE – The process of evolution and updating may be triggered by application feature additions, network service updates or user device updates.

Expected requirements

- AN-UC017-REQ-004: It is expected that autonomous networks (AN) enable the plugin of QoE prediction algorithms which may be integrated based on abstract APIs exposed from AN, which are agnostic to the type of application and the specific underlying network technology.

7.17.2 Use case specific figures

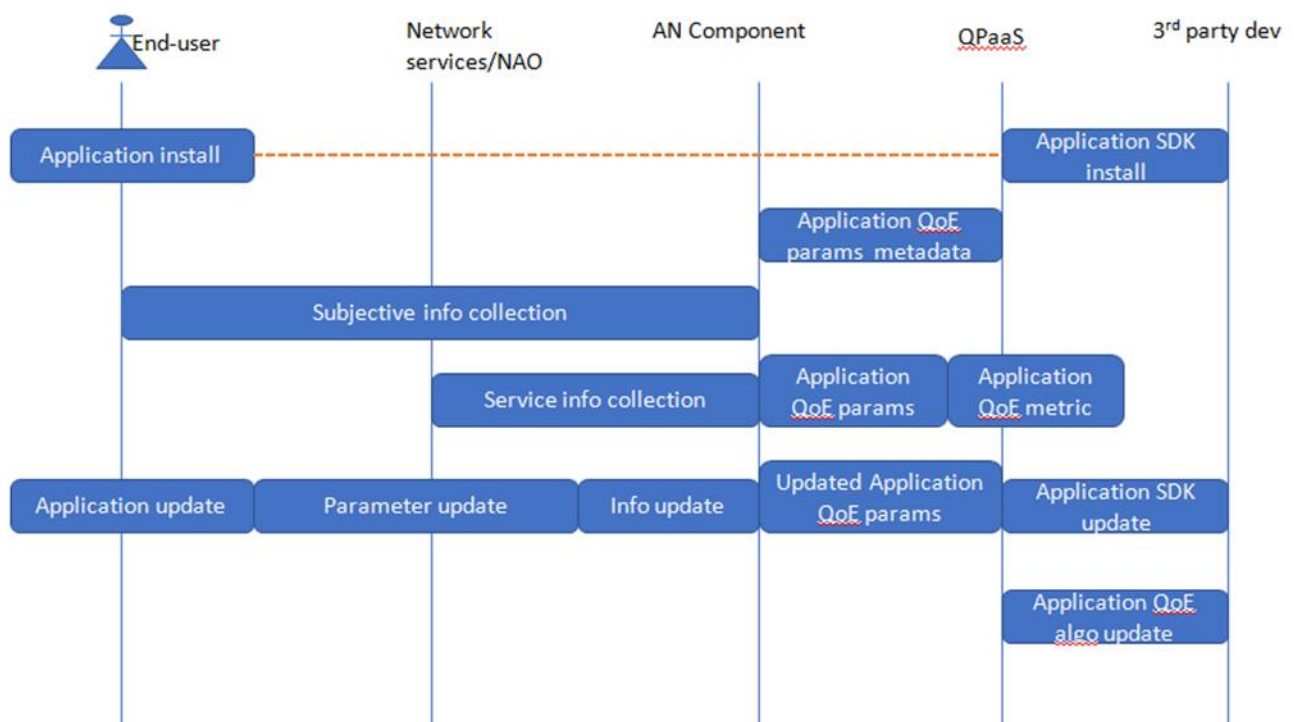


Figure 14 – Actor interaction for Quality of Experience (QoE) Prediction as-a-Service (QPaaS)

7.18 Autonomy applied to CDNs

<i>Use case id</i>	FG-AN-usecase-18
<i>Use case name</i>	Autonomy applied to CDNs
<i>Base contribution</i>	FGAN-I-079, FGAN-I-019
<i>Creation date</i>	15/April/2021
<i>Use case context</i>	Discussions regarding AN-I-019, FGAN-I-079
<i>Use case description</i>	[AN-I-019] introduced Autonomous content delivery networks (CDN), especially looking at a few key aspects of CDN and what makes them unique, focusing on several of their properties and approaches we can leverage to increase their autonomy. With increasing bandwidth of networks, proliferation in the connected

devices, increasing demand of content (e.g., live video, cloud gaming, 360 video), build-your-own approach to CDNs enabled by cloud services, cloud based CDNs are attractive but with several challenges. However caching based on data analysis remains unsolved while CDN providers struggle to provide rich content at high QoE.

[AN-I-019] called out specific aspects that need programmability- routing, caching and eviction. Importance of logging and metrics were called out on request/response metadata, timing information, internal logic decisions.

Current implementations of closed loops for managing CDNs are simplistic e.g., Standard auto scaling in the context of CDN, increasing the stream-per-node approach by hardware beef-up (better compute, more cores, more memory, L1 cache, networking including PCIe 4.0, cryptographic acceleration)

There exists an opportunity to better QoE using evolution and experimentation concepts in FG AN.

The following considerations are important to note in the context of autonomous CDNs:

1) Metric for success: CDNs usually define their success based on whether they can serve the user traffic. But this has a cyclic effect because the ability to attract traffic depends on how well CDNs process the traffic they currently have. The challenging part is the metric used to judge the CDN,

examples:

- a) Incorporating the current number of requests into the score would be useful.
- b) If there is anonymised access to user quality of experience (QoE) data, how fast the page loaded, it would be a useful metric.
- c) Another option is to measure the response time from the CDN (usually abridged to the hit ratio).
- d) Include other overlays in the measurement e.g., control planes, service management, tiered storage design.
- e) Going beyond auto scaling to healing, load balancing and edge compute, concurrency.

2) Adapting the possible caching strategies:

Memory intensive contents require large memory in CDN:

- a) Based on the treatment of various type of content and CDN use cases in the cache. e.g., live video that is cached for a brief period of time, VoD, live transcoding.

To reduce the time to live (TTL) to avoid keeping infrequently popular objects in cache

- b) Bypass the cache for large objects, or for certain classes of users, or particular extensions.
- c) Use the disk, or explicitly forbid it.
- d) Take advantage of flexibility provided by virtual cache.

3) As far as open source and standards go: (decouple the components)

- a) Open-caching is pushing to provide a subset of metrics.
- b) Perhaps an opportunity to derive "upstream" gaps in standards and lead an opensource proof of concept (PoC).
- c) Study of an open, interoperable CDN components – e.g., caching, transcoding, analytics which can help independent evolution of the CDN pipeline, while taking advantage of the work in other bodies e.g., encode/decode, AI, graphics. and hardware evolution e.g., compute/mem/network/acceleration.
- d) Similarly, take advantage of the software deployment trajectory towards cloud native.

	<p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Outer-loop: Represent the "QoS/QoE requirements" in an intent, deployment considerations (e.g., hardware, cloud) are to be captured in the intent. software/CDN pipeline considerations are to be captured here too. 2. Based on analysis, derive the cache policy, action: auto scaling/traffic routing, geographic location (e.g., edge), decide storage configurations, APIs and concurrency mechanisms. 3. Experiment to determine a good combo of KPIs, data measurement, policies, action areas (e.g., scaling, positioning) 4. Inner-loop: Adapt the CDN and corresponding configurations based on the above, with tangible, demonstrable benefits in QoE. 5. Feedback to intent evolution – to step 1 above.
<i>Open issues (as seen by the proponent)</i>	<ol style="list-style-type: none"> 1. Can we capture the app-specific constraints (e.g., hit ratio) in a standard fashion? (for inputs to perhaps other SDOs) 2. What are the traffic characteristics relevant to CDNs for 6G?
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.

7.18.1 Use case requirements

Critical requirements

- AN-UC018-REQ-001: It is critical that AN enables representation of QoS/QoE requirements in an intent, and additionally deployment considerations (e.g., hardware, cloud) and software/CDN pipeline considerations are to be captured in the intent. are to be captured here too.
- AN-UC018-REQ-002: It is critical that AN enable adaptations based on analysis.
- AN-UC018-REQ-003: It is critical that AN enable experiment to determine a good combo of KPIs, data measurement, policies, action areas (e.g., scaling, positioning).
- AN-UC018-REQ-004: It is critical that AN enable the tracing of adaptations on configurations to tangible, demonstrable benefits in QoE.
- AN-UC018-REQ-005: It is critical that AN enable feedback to intent evolution.

NOTE – Feedback may include parameters for representation in the intent, additional deployment considerations, and adaptations.

7.18.2 Use case specific figures

None.

7.19 Analysis-driven evolution in virtualized RAN based on devops

<i>Use case id</i>	FG-AN-usecase-19
<i>Use case name</i>	Analysis-driven evolution in virtualized RAN based on devops
<i>Base Contribution</i>	AN-I-072
<i>Creation date</i>	22/April/2021
<i>Use case context</i>	Discussions in the FG AN meeting during [AN-I-072]
<i>Use case description</i>	<p>Open radio access network (e.g., O-RAN) architectures allow disaggregated evolution of RAN components. Programmability and interfaces exposed by RAN components in open RANs allow developers the opportunity to create applications (e.g., xApps) based on data from RAN.</p> <p>In parallel, development methodologies like devops are being applied to enable rapid introduction of services to networks.</p> <p>At the same time, technology evolution in the form of 6G is in progress.</p>

	<p>This use case links the dev and ops cycle on one side to the programmability offered by new RAN architectures like O-RAN.</p> <p>An analysis of RAN services and applications (e.g., data, messages, interfaces, logs, etc from xApps and rApps) can provide valuable information regarding software evolution and technology evolution and deployment evolution.</p> <p>Following are related steps in this use case scenario (with O-RAN as example architecture):</p> <ol style="list-style-type: none"> 1. Analysis of heterogeneous RAN components, corresponding splits, capabilities, deployment options and interfaces and data models (e.g., E2 nodes and E2AP support). 2. Analyse the information in the near real time RAN intelligent controller (nRT RIC) 3. Discover the capabilities of various RAN nodes and instantiate (potentially cloud-native versions of) applications (e.g., xApps) based on RIC SDKs. 4. Provision and analyse the closed loops at near real time RIC. 5. In correlation with the non RT RIC, analyse the devops cycle at the near real time RIC to provision new types of CNFs in the near real time RIC and new types of E2 nodes (or new capability-needs in E2 nodes). 6. In the non RT RIC, analyse the devops cycles of near RT RIC, new capability needs of E2 nodes, arrive at new use cases (e.g., what are the users not able to do with the current network and why?) <p>Describe the relation with autonomous behaviour (if any).</p> <ul style="list-style-type: none"> – This use case is related to the evolution and experimentation aspects. It takes advantage of the increased data gathered from RAN via the open interfaces and the devops style of RIC application development to automate specific aspects of the evolution process.
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> – E2AP message structure may need to be predetermined and has to be agreed with E2 node vendors. – Unified RIC SDK space is needed.
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	<p>High</p> <ul style="list-style-type: none"> – Has the potential to impact future network architectures and use cases.
<i>References</i>	[b-DISH-AWS], [b-ONF-auto-service]

7.19.1 Use case requirements

Critical requirements

- AN-UC019-REQ-001: It is critical that AN enables analysis of heterogeneous RAN components, corresponding splits, capabilities, deployment options and interfaces and data models.

NOTE – e.g., E2 nodes and E2AP support.

- AN-UC019-REQ-002: It is critical that AN enables discovery of the capabilities of various RAN nodes and instantiate (potentially cloud-native versions of) applications.

NOTE – Examples of applications are xApps.

- AN-UC019-REQ-003: It is critical that AN enables provisioning and analysis of closed loops at near real time locations.

NOTE – Near real time RIC is an example of near real time location.

- AN-UC019-REQ-004: It is critical that AN in correlation with the orchestrator, analyse the devops cycle at the near real time locations to provision new types of network functions in the near real time locations.

NOTE – Further examples of new types of network functions are new types of E2 nodes (or new capability-needs in E2 nodes).

7.19.2 Use case specific figures

None.

7.20 Evolving edge applications for verticals using private 5G

<i>Use case id</i>	FG-AN-usecase-20
<i>Use case name</i>	Evolving Edge applications for verticals using Private 5G
<i>Base contribution</i>	[AN-I-065]
<i>Creation date</i>	22/April/2021
<i>Use case context</i>	Discussions during [AN-I-065]
<i>Use case description</i>	<p>Vertical network applications like corrosion detection and intruder detection needs to be enabled at the edge using AI/ML. These applications allow: Inspection and surveillance services for critical industrial infrastructures. Multi-domain (core and edge) e2e deployment of applications, on demand, is needed.</p> <p>5G orchestration platform allows distributed deployment of applications, especially in exploiting the capabilities at an edge environment. This allows network operators to manage the unique KPIs of services at edge sites without exposing the network architecture.</p> <p>By providing an environment to develop and deploy edge applications, to serve specific needs of verticals, network operators are able to create an ecosystem for value creation, especially for domain-focussed small businesses.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Enterprises deploy private 5G network slices at edge. 2. The applications and KPIs are analysed at the edge. 3. Network management and optimization approaches are triggered based on this analysis. 4. Tailor made applications which are specifically tuned for the needs of the enterprise are offered to the enterprise. <p>NOTE – This fits well with the concept of NetApps and network application orchestrator (NAO) [AN-I-065], decoupling the network operations logic from service provider logic and providing clear business roles.</p> <ol style="list-style-type: none"> 5. The process facilitates experimentation and evaluation of candidate solutions. 6. Edge network evolution and adaptation is triggered based on the analysis. <p>Describe the relation with autonomous behaviour (if any).</p> <ul style="list-style-type: none"> – This use case is related to the evolution and experimentation aspects. It takes advantage of the increased deployment flexibility provided by private 5G networks. edge-core information exchange is used to trigger experimentation and adaptation of the edge. – There is also an expectation of alignment with domain specific experiments and matching KPIs based on innovations in the verticals.
<i>Open issues (as seen by the proponent)</i>	– Lack of standard mechanisms for representing and experimenting with digital twin-like mechanisms for experimenting.

	<ul style="list-style-type: none"> – Lack of standards in edge-core communication for evolution and adaptation. – Lack of repository for domain-specific 3rd party applications which can be deployed at the edge. – Lack of inputs and experts from verticals.
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	<p>High</p> <ul style="list-style-type: none"> – Has the potential to impact future service deployment at the edge. – Standards gaps are visible. – Potential collaboration with EU projects.
<i>References</i>	

7.20.1 Use case requirements

Critical requirements

- AN-UC020-REQ-001: It is critical that autonomous networks (AN) interface with network application orchestration platforms at edge networks to provide both local, vertical-specific, including real time analytics as well as remote, general, including non-real time analytics.

NOTE – Network application orchestration platforms may coordinate with edge analytics and edge service management to abstract the edge network architecture to the AN.

- AN-UC020-REQ-002: It is critical that autonomous networks (AN) provide both network management and optimization and application management and optimization services to application orchestration platforms at the edge.

NOTE – While network management and optimization provides specific inputs to the edge about the network architecture, application management and optimization services may provide specific inputs on placement, functionalities and other aspects of applications.

Expected requirements

- AN-UC020-REQ-003: It is expected that autonomous networks (AN), provide tailor-made recipes for application management and optimization specific to verticals deployed at the edge.

NOTE – These recipes may be the result of offline, generalized analytics at the AN. These recipes may be considered by NAO while designing, developing and deploying applications at the edge.

Added value requirements

- AN-UC020-REQ-004: It is of added value that autonomous networks (AN), consider feedback from NAO, and continuously optimize the tailor-made recipes for application management and optimization specific to verticals deployed at the edge.

NOTE – The feedback from NAO will not contain the details of network architecture or user details at the edge.

7.20.2 Use case specific figures

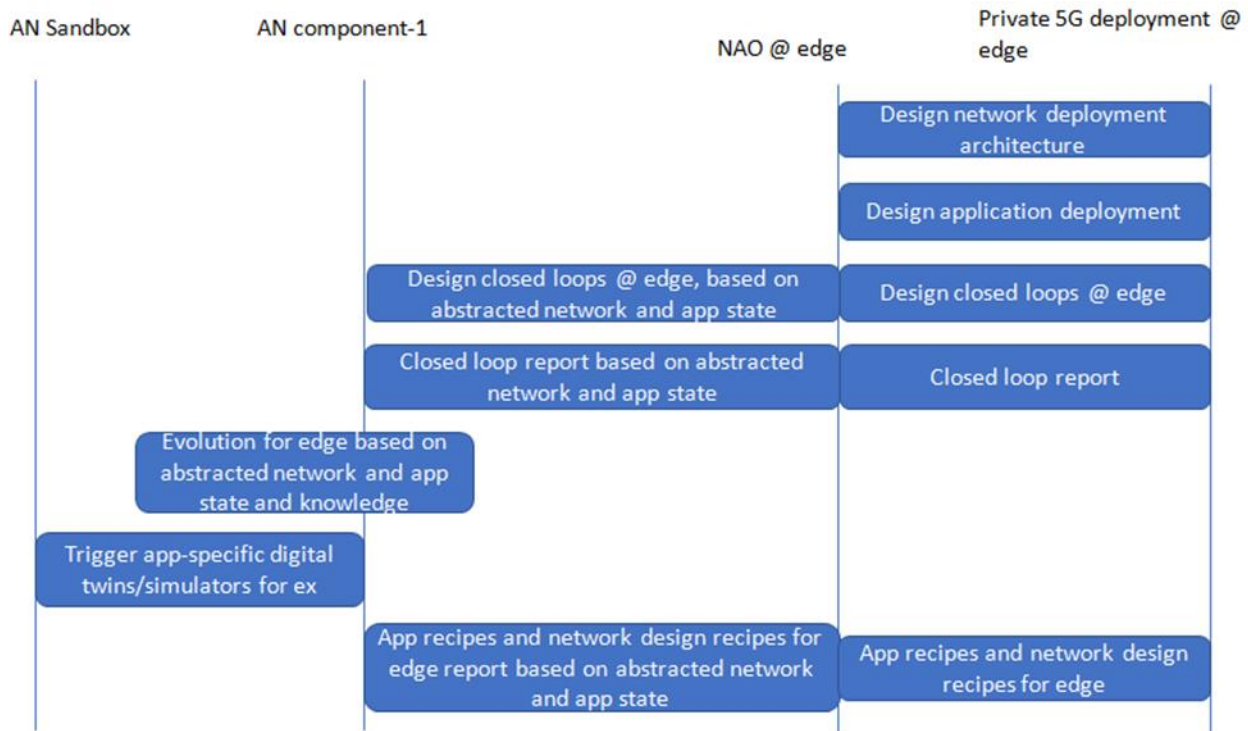


Figure 15 – Actor interaction for Evolving Edge applications for verticals using Private 5G

7.21 Experimentation and "fire-drills" for public safety networks

<i>Use case id</i>	FG-AN-usecase-21
<i>Use case name</i>	Experimentation and "fire-drills" for public safety networks
<i>Base contribution</i>	[AN-I-055-R1]
<i>Creation date</i>	22/April/2021
<i>Use case context</i>	Discussions during [AN-I-055-R1]
<i>Use case description</i>	<p>Emergency response and public safety needs resilient, on-demand network setup and management. This will require inputs from verticals including emergency responders. Experimentation and trial runs may be mandated in certain regions.</p> <p>Based on the new technologies used in evolving the networks, there are different ways of deploying public safety networks. To validate the readiness of such networks, experiments need to be designed, even for the rare scenarios. In fact for public safety networks, the design of rare scenarios is more important than the "sunny-day" success scenarios.</p> <p>It is also important that the experimentation matches step with the evolution of technologies used for implementing the public safety networks.</p> <p>Following are related steps in this use case scenario:</p> <ul style="list-style-type: none"> – Step 0: continuous analysis of external inputs and creation of strategies for experiments (experiments are equivalent to "fire drills"). – Step 1: closed loops are formed in sandboxes, "fire drills" are conducted and analysed. – Step 2: based on the "success" or failure of the rare scenarios, network optimization may be triggered.

	Describe the relation with autonomous behaviour (if any). <ul style="list-style-type: none"> – This use case is related to the evolution and experimentation aspects. It applies the principle similar to GAN for experimenting, validating the preparedness of public safety networks.
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> – Need mechanisms for representing the current state of readiness of the network to handle emergencies. – Need mechanisms for modelling and automation and validation of such controlled experiments in the context of public safety networks.
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	High <ul style="list-style-type: none"> – Addresses an important end-user scenario – Potential collaboration with other groups working in the area of public safety networks.
<i>Reference</i>	

7.21.1 Use case requirements

Critical requirements

- AN-UC021-REQ-001: It is critical that autonomous networks (AN) enable continuous analysis of external inputs and creation of strategies for experiments.
- AN-UC021-REQ-002: It is critical that autonomous networks (AN) enable closed loops formation in sandboxes, where specific tests could be conducted and analysed on those closed loops.
- AN-UC021-REQ-003: It is critical that autonomous networks (AN) trigger network optimizations based on the "success" or failure of the rare scenarios in sandboxes.

7.21.2 Use case specific figures

None.

7.22 Machine learning for network automation

<i>Use case id</i>	FG-AN-usecase-22
<i>Use case name</i>	Machine Learning for Network Automation
<i>Base contribution</i>	AN-I-014
<i>Creation date</i>	02/Feb/2021
<i>Use case context</i>	Discussions regarding AN-I-014
<i>Use case description</i>	<p>[AN-I-014] introduced Machine Learning-enabled network automation and the required "tailoring" of ML apps for networks.</p> <p>[AN-I-014] called out specific aspects that need consideration in network while applying ML: requirements from each domain, specificities (time, data and error) of each domain. Reference architecture was discussed including a ML orchestration layer. Considerations on algorithm design including trade-offs were discussed.</p> <p>The concept of "sub-problems" in networks and the limited role of ML were discussed too. The challenge in mix-match of data with ML (with respect to security, location, interoperability, etc) was discussed. A potentially top-down approach to service optimization was discussed.</p>

	<p>Following are the additional considerations in this use case:</p> <ol style="list-style-type: none"> 1. Domain specific characteristics (called "specificities" in AN-I-14) may not be known beforehand to the solution designer, especially given the loosely coupled architecture of future networks and ML. 2. In addition to domain-specificities, service based specificities may be important too. Even in case of multi-domain services, specificities could be captured per-domain, E2E, at service level. So when we add new tenants, we need a way to dynamically capture their specificities + the domains which their corresponding specificities. 3. Agile dev and deployments in future networks may need dynamic discovery of trade-offs per service. Considering the service life cycle as day-0 (design), day-1 (deployment), day-2 (monitor), day-3 (optimization), day-4 (re-design), day-5 (evolution), feedback loops are important to enable rapid development and reduce time to market for new services. 4. Run-time discovery of "sub-problems" <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. ML pipelines configure policies in the network based on the network QoS feedback. 2. Service metrics and related policies are provisioned in the ML pipelines based on the monitoring and analysis of errors. 3. Service redesign and optimization is triggered based on "sub-problems" and "specificities" discovered. 4. Network/domain specificities are tracked and similarly optimization problems are tracked. These are input to the service evolution.
<i>Open issues (as seen by the proponent)</i>	<ol style="list-style-type: none"> 1. Considering "achieve optimal operation in one domain only is risky" and "cross-tenant metrics exchange" is needed, and considering the SA5 1: many relationships with vendors in 28.530, what are the possible feedback from the CSP and CSCs? Other than the stock "metric feedback" and "policy" to the NF, is there an orthogonal feedback to "ev" to "dev" which can potentially create new NFs? 2. How do we discover "sub-problems" on the fly? 3. How to do dynamic discovery of trade-offs per service? 4. What tools we have to "design algorithm"?
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	<p>High</p> <ul style="list-style-type: none"> – Has the potential to impact future service development and evolution. – Reuses MLFO and Y.3172 architecture.
<i>Reference</i>	

7.22.1 Use case requirements

Critical requirements

- AN-UC022-REQ-001: It is critical that autonomous networks (AN) enable, in case of multi-domain services, specificities per-domain, E2E.

NOTE – This includes a way to dynamically, autonomously capture their specificities + the domains which their corresponding specificities.

- AN-UC022-REQ-002: It is critical that AN enable agile dev and deployments in future networks dynamic discovery of trade-offs per service.

NOTE – Considering the service life cycle as day-0 (design), day-1 (deployment), day-2 (monitor), day-3 (optimization), day-4 (re-design), day-5 (evolution), feedback loops are important to enable rapid development and reduce time to market for new services.

- AN-UC022-REQ-003. it is critical that AN enable run-time discovery of "sub-problems".

7.22.2 Use case specific figures

None.

7.23 Autonomous agents (with varied competence) in networks

<i>Use case id</i>	FG-AN-usecase-23
<i>Use case name</i>	Autonomous agents (with varied competence) in networks
<i>Base contribution</i>	FGAN-I-052_att
<i>Creation date</i>	04/Mar/2021
<i>Use case context</i>	Discussions regarding FGAN-I-052_att
<i>Use case description</i>	<p>[AN-I-052] introduced autonomous Systems in hostile environments. Estimation and judgement of competence as key criteria for determining the right level of autonomy was the focus. Levels of autonomy for unmanned systems were introduced, especially ranging from "sub-functions" to single functions to single system to teams. A framework for robot autonomy was discussed with corresponding guidelines. Based on these, the characteristics for operations in hostile environments were listed.</p> <p>[AN-I-052] called out the requirements for operations with autonomous systems. Relation between the needed level of autonomy depending on environment and type of task in contrast to capabilities of the system in combination with policies. Systems should provide the best possible support and hence the autonomy level has to be adjusted such that humans only have to intervene when it is necessary and makes sense. Due to dynamics during missions in hostile environments, the systems have to adapt their autonomy level at run-time (or humans have to do it) according to the situation and the corresponding requirements. The existing stage levels of autonomy have to be extended by the ability of switching the level in run-time (judgement by design vs. judgement in run time)</p> <p>A simple workflow scheme for autonomy with varying autonomy levels was discussed. This included task specification by human and task understanding, feasibility check, task planning, task execution by system. Request for support to human and control by human can be to any of these workflow steps. Monitoring of performance levels by humans and learning by the system are added steps. Autonomy level and dependency on competence were discussed. Competence analysis as a weighted function of capabilities needed, capabilities existing, existing options for actions and existing constraints was described.</p> <p>Following are the additional considerations in this use case:</p> <ol style="list-style-type: none"> 1. Taking telco service design, development, deployment and operations (ops) as an example – the levels of autonomy may be applied in follows: <ol style="list-style-type: none"> a) Service design is done by designers (100% designer interaction, no ops interaction). b) Existing software development kits (SDKs) and application programming interfaces (APIs) are exercised to create applications (e.g., rApps or xApps) – (high level designer interaction, high code but low ops interaction). c) Configuration of existing or new services in real time environments (e.g., distributed unit (DU) – mid level designer interaction, high ops involvement). d) Service deployment and QoE measurement in customer premises (e.g., VoD – no code, low designer interaction, collaborative, high ops involvement). <p>Thus, it may be relevant to consider the nature of the task in addition to the type of environment. e.g., for the ops engineers design phase is a "difficult</p>

	<p>environment" (due to low involvement) and for the service designers customer premises is a "difficult environment" (due to constraints in site visits). Thus, it may be relevant to consider a multi-agent system where the agents have varied competences (capabilities + options for actions + constraints). Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Problem detected in the network (e.g., video performance degradation for customers) <ul style="list-style-type: none"> – e.g., scenario: ops agents collect debug data. analysis agents with matching capabilities are deployed at the nRT RIC and triggered to analyse the data. 2. Fault isolation (e.g., in CU, DU, user plane, control plane) <ul style="list-style-type: none"> – e.g., scenario: collaborative analysis is used to pin-point the cause of failure. this may involve multi-agent team communication to do the steps in the workflow described in AN-I-052. 3. Fault correction (e.g., parameter configuration) <ul style="list-style-type: none"> – e.g., scenario: may involve service upgrade or software reconfig. 4. Analysis of task performance by agents <ul style="list-style-type: none"> – e.g., scenario: collaborative analysis is used to collect data from the agents – including where the human interactions were needed. 5. Trigger creation of new agents with new capabilities. <ul style="list-style-type: none"> – e.g., scenario: location and capabilities are selected based on the next higher level of autonomy to reduce human interaction in this scenario.
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> – Need coordination of multi-agent systems to avoid conflicts. – Need mechanisms for adapting autonomy based on competence at run-time and flagging human (or higher level) assistance. – Need multi-domain orchestration to position the agents at various network domains. – In some cases, hardware capability adaptation may be needed which may need new physical network functions (PNF).
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	High <ul style="list-style-type: none"> – Has the potential to impact future service development and evolution.
<i>Reference</i>	[b-Beyerer], [b-Hesse]

7.23.1 Use case requirements

Critical requirements

- AN-UC023-REQ-001: It is critical that autonomous networks (AN) enable, autonomous agents to collect debug data.

NOTE – For example analysis agents with matching capabilities are deployed at the nRT RIC and triggered to analyse the data.

- AN-UC023-REQ-002: It is critical that autonomous networks (AN) enable, fault isolation (e.g., in CU, DU, user plane, control plane) using collaborative analysis is used to pin-point the cause of failure.

NOTE – This may involve multi-agent team communication to do the steps in the workflow.

- AN-UC023-REQ-003: It is critical that autonomous networks (AN) enable, fault correction (e.g., parameter configuration) using service upgrade or software reconfig.
- AN-UC023-REQ-004: It is critical that autonomous networks (AN) enable analysis of task performance by agents.

NOTE – Collaborative analysis is used to collect data from the agents – including where the human interactions were needed.

- AN-UC023-REQ-005: It is critical that autonomous networks (AN) enable creation of new agents with new capabilities.

NOTE – Example location and capabilities are selected based on the next higher level of autonomy to reduce human interaction in this scenario.

7.23.2 Use case specific figures

None.

7.24 Automated, adaptive acceleration for AI @ edge

<i>Use case id</i>	FG-AN-usecase-24
<i>Use case name</i>	Automated, adaptive acceleration for AI @ edge
<i>Base contribution</i>	FGAN-I-046
<i>Creation date</i>	04/Mar/2021
<i>Use case context</i>	Discussions regarding FGAN-I-046
<i>Use case description</i>	<p>[AN-I-046] introduced Spatial Architectures which scale performance & resources to meet the application requirements and Scaling to fit into available resources – in the context of DNN. It also discussed that reduced precision can be highly effective to reach communication requirements. Spatial architectures can exploit custom arithmetic at a greater degree. Further, it discussed topologies fully co designed for hardware architecture, where the Circuit is the DNN [b-Umuroglu]. Adjust the parameters of DNN (=lookup table (LUT) contents) while iterating on training dataset until accuracy.</p> <p>[AN-I-046] showed results (with an example of intrusion detection) that spatial processing, customized arithmetic and learned circuits can help scale to communication throughput and latency requirements.</p> <p>I-046 also talked about [FINN-R] and Providing tools and platforms for exploration of DNN compute architectures. ML engineers can create specialized hardware architectures on an FPGA with spatial architectures and custom precision. Design and runtime software tools (e.g., FINN) for DNN to FPGA development starting with training or learning reduced precision DNNs, using ONNX based intermediate representation, perform optimization on this intermediate representation, to create a DNN hardware IP, was discussed.</p> <p>Thus, it may be relevant to consider the following aspects for this specific use case:</p> <ol style="list-style-type: none"> 1) AI-enabled applications are increasingly being deployed at the edge. Low latency, low power consumption and small footprint are considerations for AI applications at the edge. Accelerated, AI-enabled applications at the edge are important enablers for future networks. 2) As AI technology evolves, AI models evolve, the acceleration platform must also be adaptable and at the same time satisfying the requirements above. Also, reduced time to market, development time and cost to reach production readiness, are important factors influencing deployment decisions by network operators. fully customized circuit board is developed for each application may not fit this bill. 3) Pluggable solutions into a larger edge application, providing both the flexibility of a custom implementation with the ease-of-use and reduced time-to-market of an off-the-shelf solution, are needed. 4) Adaptive computing includes hardware that can be highly optimized for specific applications such as Field Programmable Gate Arrays (FPGAs). In addition to FPGAs, new types of adaptive hardware such as adaptive

	<p>System-on-Chip (SoC) which contains FPGA fabric, coupled with one or more embedded CPU subsystems, have been introduced recently.</p> <p>5) Prebuilt platforms and APIs, software tools enable full customization of the adaptive hardware, enabling even more flexibility and optimization. This can be used to design highly flexible, yet efficient systems at the edge.</p> <p>6) Exploiting the development and adoption of standards in interface and protocols at the edge, different AI-enabled edge applications can use similar hardware components.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Given an AI/ML model layered architecture, the following considerations needs to be applied – (a) concurrency in processing of layers, (b) fragmentation/buffering between layers vs. offloading of layers into compute, (c) precision vs. performance and energy efficiency. 2. Given the specific goals and constraints of the AI/ML model, consider the target platform architecture and "sacrificable" precision to explore the model architecture and layer compositions. 3. Transformation of an AI/ML model -> intermediate representation -> optimization -> hardware implementation -> evaluation -> back to training/modelling. 4. Derive feedback for hardware adaptation and design.
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> – The role played by MLFO is not clear (need further study). – The intersection with ITU-T Y.3179 has to be studied further. – Dependencies with specific hardware architectures and specific types of neural network (NN) topologies need further study. – The role played by training techniques need further study. – The role played by cycle-faithful simulators like [scalesim] has to be studied further.
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	<p>High</p> <ul style="list-style-type: none"> – Has the potential to impact AI/ML operations and optimization in relation with hardware evolution.
<i>Reference</i>	[b-Blott], [b-Xilinx], [b-Umuroglu], [b-SCALE-Sim]

7.24.1 Use case requirements

Critical requirements

- AN-UC024-REQ-001: It is critical that autonomous networks (AN) enable, analysis of concurrency in processing of layers in a DNN, fragmentation/buffering between layers vs. offloading of layers into compute and analysis of precision vs. performance and energy efficiency.
- AN-UC024-REQ-002: It is critical that autonomous networks (AN) consider the target platform architecture and "sacrificable" precision to explore the ML model architecture and layer compositions.
- AN-UC024-REQ-003: It is critical that autonomous networks (AN) enable transformation of a AI/ML model -> intermediate representation -> optimization -> hardware implementation -> evaluation -> back to training/modelling.
- AN-UC024-REQ-004: It is critical that autonomous networks (AN) derive feedback for hardware adaptation and design.

7.24.2 Use case specific figures

None.

7.25 Assistive networks: Adaptation of communication system based on changing user accessibility needs

<i>Use case id</i>	FG-AN-usecase-25
<i>Use case name</i>	Assistive networks: Adaptation of communication system based on changing user accessibility needs.
<i>Base contribution</i>	Editor, Discussions about common user profile (CUP)
<i>Creation date</i>	10/May/2021
<i>Use case context</i>	Discussions about common user profile (CUP)
<i>Use case description</i>	<p>This contribution attempts to describe a use case where:</p> <p>For both disabled and the able-bodied, as the accessibility requirements evolve, adaptations need to be applied on the network, device and user profiles need to align with the changing needs of the user.</p> <p>The scope of assistive technologies needs to be broadened to "assistive networks". Assistive networks can be thought of E2E network slices that include assistive, adaptive, and rehabilitative connectivity for persons with specific needs. It also includes the automated mechanisms used in selecting, locating, using and customizing the networks. Assistive networks promote greater independence by enabling people to connect to the devices and network more autonomously.</p> <p>Environmental models exist for surroundings but building on top of such environment models to adapt the connectivity to the user with specific requirements is the need of the hour.</p> <p>User model and simulations are needed to provide inputs to AN.</p> <p>Development of standard definitions are needed for application model, assistive network, context modelling, environmental model and common user profile, metadata, simulation and virtual instance in the context of AN.</p> <p>Individualization needs to be added as an important dimension of AN for future networks.</p> <p>Reuse of common user profile (CUP) to automate the collection, analysis and adaptation of the network and applications is proposed here.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Environment model including the network environment is built for the user. <ul style="list-style-type: none"> – e.g., radio propagation models, signal strengths with respect areas, mobility prediction models. 2. User model is accessed and updated. <ul style="list-style-type: none"> – e.g., user specific constraints, user inside a car wearing seat belt has limited mobility within the car. Similarly for elderly and children, persons under emergency needs. 3. Simulations are used (offline and/or real time) to determine the changes and adaptations needed in the network to satisfy the needs of the user. <ul style="list-style-type: none"> – e.g., digital twins which include environment simulations and user specific criteria. 4. Adaptations are applied to the network and the context. <ul style="list-style-type: none"> – e.g., drone based coverage is provided, reconfigurable intelligent surface (RIS) configurations or beam configurations to provide better coverage. 5. Generalizations and evolutions are studied for applicability in a larger context. <ul style="list-style-type: none"> – e.g., continuous update of models, transfer of model parameters across domains for easy learning, evolution of network simulators and context for new encountered scenarios.

	Describe the relation with autonomous behaviour (if any). <ul style="list-style-type: none"> – This use case is related to adding environment sensing and adaptation to include inclusivity in the evolution and experimentation aspects. It takes advantage of the existing studies in Q11/9.
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> – Standard mechanisms for capturing environment in the context of network requires further study. – Simulations and experiments in relation to user model, specific user needs and network characteristics needs further study.
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	High <ul style="list-style-type: none"> – Has the potential to evolve assistive future network architectures and further enable new use cases.
<i>References</i>	[b-J.acc-us-prof], [b-Biswas], [b-AVA-1], [b-AVA-2], [b-ISO/IEC 24756], [b-ISO 9241-129], [b-KUKA]

7.25.1 Use case requirements

Critical requirements

- AN-UC025-REQ-001: It is critical that autonomous networks (AN) enable, creation and representation of environment model including the network environment for the user with assistive needs.

NOTE – e.g., radio propagation models, signal strengths with respect areas, mobility prediction models.

- AN-UC025-REQ-002: It is critical that autonomous networks (AN) enable updation of user model.

NOTE – e.g., user specific constraints, user inside a car wearing seat belt has limited mobility within the car. Similarly for elderly and children, persons under emergency needs.

- AN-UC025-REQ-003: It is critical that autonomous networks (AN) enable Simulations (offline and/or real time) to determine the changes and adaptations needed in the network to satisfy the needs of the user.

NOTE – e.g., digital twins which include environment simulations and user specific criteria.

- AN-UC025-REQ-004: It is critical that autonomous networks (AN) enable Adaptations applied to the network and the context.

NOTE – e.g., drone based coverage is provided, reconfigurable intelligent surface (RIS) configurations or beam configurations to provide better coverage.

- AN-UC025-REQ-005: It is critical that autonomous networks (AN) enable generalizations and evolutions which are studied for applicability in a larger context.

NOTE – e.g., continuous update of models, transfer of model parameters across domains for easy learning, evolution of network simulators and context for new encountered scenarios.

7.25.2 Use case specific figures

None.

7.26 Ev-as-a-service: Achieving zero touch evolution in a delegated autonomy case

<i>Use case id</i>	FG-AN-usecase-26
<i>Use case name</i>	Ev-as-a-service: Achieving zero touch evolution in a delegated autonomy case.
<i>Created by</i>	Editor
<i>Mentor</i>	Laurent Ciavaglia
<i>Creation date</i>	10/May/2021

<i>Use case context</i>	Discussions during and after FGAN-I-005
<i>Use case description</i>	<p>This contribution attempts to describe a use case where:</p> <ul style="list-style-type: none"> – A multi-domain architecture is assumed. Each domain may have its own orchestrator. NOTE – Example in FG-AN-usecase-20, NAO is mentioned. MANO/NFVO is used in the NFV domain, service orchestrator may be used similar to ONAP in the communication domain. – Closed loops are assumed in each domain, managed by the corresponding orchestrators. – (For the purposes of this use case), it is assumed that each closed loop enables autonomous behaviour in that domain for specific use cases e.g., resource scaling based on load. NOTE – The autonomous behaviour enabled by use case specific closed loops and managed by domain orchestrators can be extended to any number of management domains. – Current frameworks [ITU-T Y.3172], [ITU-T Y.3179], [b-ETSI ZSM 002] assume offline development and provisioning of services which form the closed loops. e.g., AI/ML model training based on data from the network, followed by model serving in the network. – This use case introduces an evolution (Ev) function which analyses the inputs from the closed loops (and other context information in the domain orchestrator) to trigger creation of new services which can cater to the evolving needs of the domains. – The triggers may be input to devops pipeline. This may result in creation of new framework services or new applications or new VNFs or new configurations or new AI/ML models etc. – These may then be tested and evaluated in an experimental setup (e.g., Digital twins, Sandbox, etc) and deployed in corresponding domains using the domain orchestrators. <p>In summary, the use case proposes to monitor, identify the need for ev, generate new f() to support this need, "(re-)inject" that function through devops pipeline into the closed loops. Note that this may require multi-domain coordination to modify the closed loops and may be challenging from an implementation perspective. Implementation may depend on the capabilities provided by the underlying closed loop frameworks e.g., ZSM.</p> <p>Levels of "mutation" of CL:</p> <p>There can be a spectrum of adaptation changes to the closed loops (CL):</p> <ol style="list-style-type: none"> a) No adaptation at all – same input to the CL, always leads to the same output. b) Limited adaptation – CL improves utility over time, so same input may not lead to the same output (after improvement). c) Full-fledged evolution, involving development and injection of new functions. <p>The capability of underlying closed loop frameworks may be factor in deciding the level of adaptation possible in the AN.</p> <p>Division of responsibility between the controller and the CL:</p> <ol style="list-style-type: none"> 1) "Dumb" vs. "intelligent" closed loops: dumb CL may allow full fledged re-configurations and re-injections of functions which may allow overall mutation of its functionality over time. Whereas an "intelligent" closed loop may use its intelligence to limit external influence by the controller. In any case, the domain orchestrator should know the mutation-capabilities of the CL. In case of limitations encountered for adaptations, the CL (or domain orchestrator) should be able to escalate the requirement to higher domains.

	<p>2) Timescale of Ev has to be agreed between the CL and the orchestrator. Following are related (example) steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Enterprise/vertical provides intent for application/service. 2. A corresponding slice is created by NAO. 3. Corresponding resources are allocated by NFV MANO. 4. NSaaS may be instantiated using ONAP/SO. 5. Use case specific closed loops are instantiated in each domain e.g., power optimization, interference management, resource utilization, self-x. 6. Ev as a Service is instantiated in the zero touch framework. 7. Based on the analysis of inputs from use case specific closed loops and domain orchestrators, Ev triggers configurations, updates, service instantiation, new closed loops, even new service development (using triggers to Devops pipeline). 8. After testing and validation, such updates are reflected in the domains and use case specific closed loops. <p>Describe the relation with autonomous behaviour (if any).</p> <ul style="list-style-type: none"> – This use case proposes Ev-as-a-service scenario in relation to zero-touch frameworks.
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> – Interfaces between NAO and other orchestrators are to be studied. – Interface with devops pipeline is not automated (as of now). – Human interaction in the process is to be studied – Whether AI/ML services need special treatment in this process? – What supporting enablers are needed to enable Ev? How do we evaluate whether the proposed Ev meets the KPIs?
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	<p>High</p> <ul style="list-style-type: none"> – Enables integration and interaction with zero touch frameworks.
<i>References</i>	[b-ETSI GS ZSM 001], [b-ETSI GS ZSM 002], [b-ETSI GR ZSM 009-3], [b-Ciavaglia-1], [b-ETSI GS ZSM 013], [ITU-T Y.3172], [ITU-T Y.3179]

7.26.1 Use case requirements

Critical requirements

- AN-UC026-REQ-001: It is critical that evolution function (Ev) in autonomous networks (AN) analyses the inputs from domain specific closed loops (and other context information in the domain orchestrator) to trigger management (creation, update and delete) of network services, which may in turn participate in the closed loops.

NOTE – The management (creation, update and delete) of network services, over a number of iterations, may result in evolution.

- AN-UC026-REQ-002: It is critical that the modifications to network services and applications may be tested and evaluated in an experimental setup and deployed in corresponding domains using the domain orchestrators.

NOTE – Examples of experimental setups used for testing and evaluation of network services and applications are Digital twins, Sandbox, etc.

Expected requirements

- AN-UC026-REQ-003: It is expected that management of network services or applications or VNFs or configurations or AI/ML models is done at runtime in coordination with devops pipelines.

- AN-UC026-REQ-004: It is expected that domain specific closed loops allow management of network services or applications or VNFs or configurations or AI/ML models in coordination with AN components.

NOTE – There can be a spectrum of adaptation changes (levels of "mutation") of network services:

- a) No adaptation at all.
- b) Limited adaptation – improves utility over time.
- c) Full-fledged evolution, involving development and injection of new functions.

The capability of underlying closed loop frameworks may be factor in deciding the level of evolution and adaptation possible in the AN.

Added value requirements

- AN-UC026-REQ-005: It is of added value that Ev function in autonomous networks (AN) act as consumer of mutation functions provided by underlying service management frameworks and in turn provides evolution service to underlying service management frameworks.

NOTE – Example of a service management framework is ETSI ZSM [b-ETSI GS ZSM 001].

7.26.2 Use case specific figures

None.

7.27 Experimentation as a service: Digital twins as platforms for experimentation

<i>Use case id</i>	FG-AN-usecase-27
<i>Use case name</i>	Experimentation as a service: Digital twins as platforms for experimentation.
<i>Base contribution</i>	FGAN-I-058
<i>Creation date</i>	10/May/2021
<i>Use case context</i>	Discussions during presentation of FGAN-I-058
<i>Use case description</i>	<p>[FGAN-I-058] described a digital twin as a representation of a physical and/ or logical object. The contribution proposed to build Digital Twins of Computer Network infrastructures. Some examples of the (hypothetical) questions which could be answered using digital twins were listed as: Which is the best network upgrade given a budget? Which is the best link upgrade to accommodate a new customer? Can we support a new customer SLA with the current network capacity? etc.</p> <p>The impact of digital twins in Network Planning and Upgrading, Troubleshooting and Performance Analysis, What-if Analysis were described in [FGAN-I-058]. This makes digital twins a perfect environment for experimentation in the context of autonomous networks.</p> <ol style="list-style-type: none"> 1. [FGAN-I-058] took the approach of using neural networks (NN) to build digital twins. The approach using graph neural networks (GNN) was described. It generalizes to unseen topologies, routings and traffics. Specific example of RouteNet was described. RouteNet can generalize to unseen topologies, routings and traffic matrices. 2. DRL+GNN looks as a promising technique for real-time network optimization was introduced in [FGAN-I-058]. <p>Autonomous networks (AN) aim to remove the human from the control loop. This poses hard challenges to offer 100% guarantees once the AN products are deployed in networks. In order to achieve mature solutions for autonomous network control, it will be essential for AN vendors to validate in advance that their products will operate successfully in the target customer networks, before they are actually deployed.</p> <p>A DT can be used to estimate accurately the resulting network performance of an experimentation approach and the effect after applying the actions produced</p>

	<p>by the AN, thus determining what network scenarios are well-supported by the product. After a comprehensive validation test, the vendor can apply the adaptations to the network.</p> <p>Following are examples of scenarios in this use case:</p> <p>Scenario-1: preparation of DT: import network configurations (may include closed loops?) into digital twin. This sets the stage for preparation of simulations in the DT.</p> <p>Scenario-2: trigger of DT for simulations: update of network configurations in digital twin (if any, by engineer), followed by simulations in digital twin and generation of asynchronous events. These events are consumed by AN engine and may in turn result in experimental configurations/updates from the AN engine towards the DT. This cycle may continue based on the sequence of simulations and scenarios in the DT. The validation of KPIs in the DT as a result of experimentation and adaptations by the AN engine is an important step.</p> <p>Scenario-3: trigger of AN engine by operator: update of network policy/configurations by engineer, which triggers AN engine to corresponding experiments or configurations towards the digital twin. Experimentation may be configured in the digital twin and corresponding events and KPIs may be used to evaluation the result of the experimentation. This may result in selecting the best possible sequence of actions or adaptations towards the network. The validation of the AN engine actions is an important step here.</p> <p>Scenario-4: trigger based on evolution: Other triggers for experiments in AN engine may include inputs from evolution functionality. Experimentation and evaluation of actions or adaptations towards the network are same as above.</p> <p>Following are related steps in this use case scenario:</p> <ol style="list-style-type: none"> 1. Import environment into DT, trigger simulations in DT and validate the results, especially the use case specific closed loops. 2. AN-triggered experiments and adaptations are tested using corresponding simulator settings in DT and evaluating the impact in simulations. <p>Describe the relation with autonomous behaviour (if any).</p> <ul style="list-style-type: none"> – This use case is related to the concept of experimentation.
<p><i>Open issues (as seen by the proponent)</i></p>	<ul style="list-style-type: none"> – What are the features of the digital twin? what are the APIs that the digital twin expose? – How to configure closed loops in DT? – We discussed "a marketplace of DTs and you choose the one that fits your particular needs" – this needs further study. – If a DT is trained with certain capabilities, can we transfer the training using p2p interface to new DT? – How to take advantage of GNNs for generalisability and explainability? – Are there standard interfaces for integrating AI/ML with digital twin? Is Y.3172 applicable here?
<p><i>Notes on use case category</i></p>	<p>Cat 2: Describes a scenario related to application of autonomous behaviour in the network.</p>
<p><i>Notes on priority of the use case</i></p>	<p>High</p>
<p><i>References</i></p>	<p>[b-Rusek], [b-Almasan]</p>

7.27.1 Use case requirements

Critical requirements

- AN-UC027-REQ-001: It is critical that AN enable import of simulation environment into DT, trigger simulations in DT and validate the results, especially the use case specific closed loops.
- AN-UC027-REQ-002: It is critical that AN-triggered experiments and adaptations are tested using corresponding simulator settings in DT and the impact in simulated environment is evaluated.

7.27.2 Use case specific figures

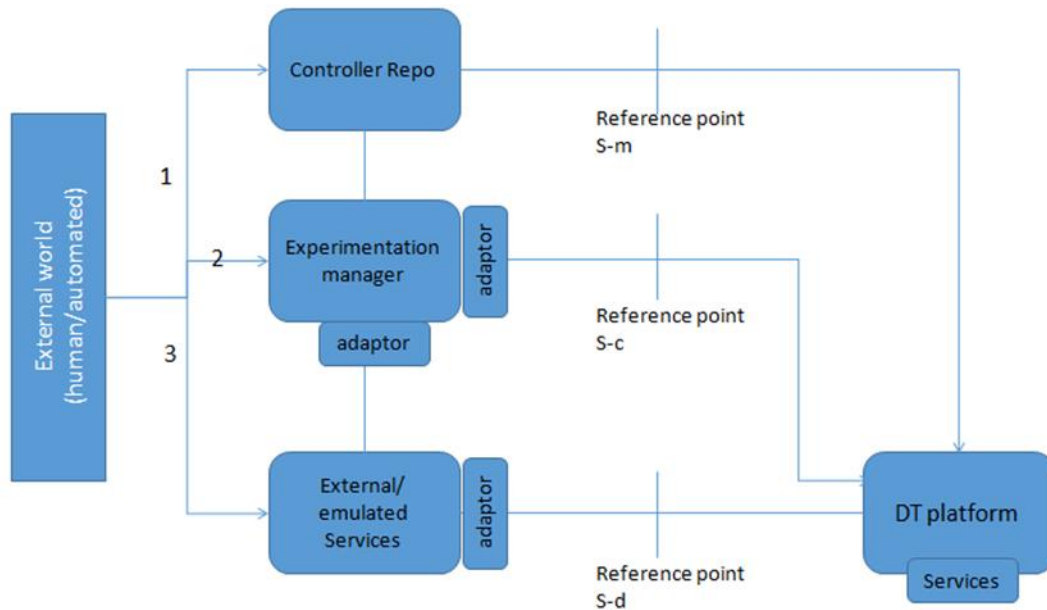


Figure 16 – Experimentation as a service

7.28 Evolution from scenario-specific, explicit-coordination to coordination-free interoperability (achieved using data-driven approaches)

<i>Use case id</i>	FG-AN-usecase-28
<i>Use case name</i>	Evolution from scenario-specific, explicit-coordination to coordination-free interoperability (achieved using data-driven approaches)
<i>Base contributions</i>	[FGAN-I-091], [FGAN-I-029]
<i>Creation date</i>	27/May/2021
<i>Use case context</i>	Discussions during [FGAN-I-091], [FGAN-I-029]
<i>Use case description</i>	<p>Deep learning is being used to address challenging problems in wireless communications such as modulation recognition, radio fingerprinting and many other scenarios. The advantages of this approach include the capability to address wide-range of scenarios, where a mathematical model is difficult to make (e.g., channel estimation, beam management for future networks).</p> <p>Existing solutions mostly rely on explicit coordination between the transmitter (TX) and the receiver (RX), introducing problems of interoperability and necessity for standards. Such signalling messages eat into the costly spectrum and complicate protocol design.</p> <p>A data-driven approach based on neural networks (NN) is an alternative to achieve coordination-free interoperability.</p> <p>Example-1 for the use case:</p>

Millimeter wave (mmWave) communication with large antenna arrays is a promising technique to enable extremely high data rates. Because of their highly directional transmissions, radios operating at millimeter wave (mmWave) frequencies need to perform beam management to establish and maintain reliable mmWave links. Transmitter (TX) and the receiver (RX) need to coordinate to select the beam pair that yields the highest beamforming gain. Currently 5G NR defines the stages of exhaustive beam sweep (EBS) as: initial access (IA) and beam tracking. For both IA and beam tracking, the 3rd Generation Partnership Project (3GPP) NR standard for 5G communications utilizes synchronization signal blocks (SSBs).

Beam management for the IA procedure in 3GPP NR involves:

- 1) Beam sweep: base station transmits directional Synchronization Signals (SSs) to cover all the TXBs of a certain codebook. Each beam is swept with an SSB, which is a group of 4 OFDM symbols and 240 subcarriers in frequency.
- 2) Beam measurement: the User Equipment (UE) itself, if configured for directional reception, performs a directional scan, measuring the quality of each beam pair
- 3) Beam decision: the UE selects the beam to be used to perform initial access
- 4) Beam reporting: During the next SSB in the selected direction, the UE acquires information on the time and frequency resources in which the base station will be in receive mode for the random access message using the same TXB

I-091 explained receiver can associate Signal-to-Noise-Ratio (SNR) levels to beams without explicit coordination with the transmitter. pilot-less estimation technique. the RX infers the Angle of Arrival (AoA) and the TXB by passively eavesdropping on data transmissions to other users in the network. leveraging a data-driven approach based on convolutional neural networks (CNNs) to achieve coordination-free beam management in mmWave networks. based on a unique "signature" of the beam from the impairments.

Inferring (i) the Angle of Arrival (AoA) of the beam and (ii) the actual beam being used by the transmitter through waveform-level deep learning on ongoing transmissions between the TX to other receivers.

Experimentation: experimental data collection campaign with two software-defined radio testbeds, and by using multiple antennas, codebooks, gains and locations (iv) 3 different AoAs; (v) multiple TX and RX locations.

Evaluation criteria: An upper bound on the expected search time of the proposed algorithm. The proposed technique reduces latency by up to 7x with respect to the 5G NR initial beam sweep in a default configuration and with a 12-beam codebook.

Following are related steps in this use case scenario:

1. Based on the analysis of data from the network, reference points are selected by Ev() where data-driven NN based approaches can be applied to reduce signalling. Ev() should cherry-pick the reference points which has the best trade-offs in terms of benefits (e.g., spectral efficiency, latency, etc) as against the cost of training. Ev() should also help in understanding the experimentation approaches to follow.
2. Based on the scenario under study (for evolution), experimentation is setup and data sources are provisioned and ML pipelines are setup [ITU-T Y.3172].
3. Based on the evaluation of the AI/ML models in the sandbox, they are injected into the network functions (NFs).
4. Control and data flows are modified according to the evolved network.

Relation with autonomous behaviour -

<i>Open issues (as seen by the proponent)</i>	
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behavior itself.
<i>Notes on priority of the use case</i>	High – Has the potential to impact future network architectures and use cases.
<i>Reference</i>	[b-O'Shea-1], [b-O'Shea-2], [b-Jagannath], [b-Mao], [b-AI4Good-1], [b-AI4Good-2], [b-AI4Good-3]

7.28.1 Use case requirements

Critical requirements

- AN-UC028-REQ-001: It is critical that AN enable selection of reference points are based on evolution where data-driven NN based approaches can be applied to reduce signalling.

NOTE – Ev() should cherry-pick the reference points which has the best trade-offs in terms of benefits (e.g., spectral efficiency, latency, etc) as against the cost of training. Ev() should also help in understanding the experimentation approaches to follow.

- AN-UC028-REQ-002: It is critical that AN enable, based on the scenario under study (for evolution), experimentation setup and data sources provisioning and ML pipelines setup [ITU-T Y.3172].
- AN-UC028-REQ-003: It is critical that AN enable, injection of ML models into the network functions (NFs), based on the evaluation of the AI/ML models in the sandbox,
- AN-UC028-REQ-004: It is critical that AN enable, modification of Control and data flows according to the evolved network.

7.28.2 Use case specific figures

None.

7.29 Intelligent maintenance assistance system

<i>Use case id</i>	FG-AN-usecase-029
<i>Use case name</i>	Intelligent Maintenance Assistance System
<i>Base contribution</i>	FGAN-I-108
<i>Creation date</i>	24/June/2021
<i>Use case context</i>	Discussions in the weekly meeting during presentation of FGAN-I-108
<i>Use case description</i>	<p>The intelligent maintenance assistance system is an intelligent service system for network operation and maintenance. The system combines AI algorithms and AR capabilities to provide intelligent assistance for front line staff of operators in aspects of network operation and maintenance.</p> <p>The system includes backstage support system and AR glasses app. The backstage support system is deployed in cloud servers in the form of micro-service, and is connected with the network management system of operators to exchange data. The AR glasses app is used for staff's on-site work.</p> <p>The backstage support system provides the following functions.</p> <ol style="list-style-type: none"> 1. AI algorithms and AR capabilities for system. Developers can also use these algorithms to develop applications. <ol style="list-style-type: none"> 1) AI algorithms include Bar code / QR code recognition, OCR, device port recognition. 2) AR capabilities include image recognition and tracking, 3D object recognition and tracking, visual simultaneous localization and mapping (SLAM).

	<p>2. The data management function can help implement equipment data transmission, storage, and management, including network resource data, equipment status data, equipment operation data, etc.</p> <p>3. The system management function includes user management, authority management, system operation management, parameters configuration and other functions to ensure the stable and reliable operation of the system.</p> <p>The AR glasses app provides the following functions.</p> <p>1. Information collection. Staff can collect pictures through the camera of AR glasses and upload them to the backstage support system. The AI algorithms in the backstage support system can recognize the collected pictures and save the recognition results. Here, the staff can collect the text information of the labels of devices to supplement and update the information in the backstage support system.</p> <p>2. Data visualization. The data managed in the backstage support system can be displayed on the AR glasses. This function can provide the display of alarm information, base station information, equipment information, electricity consumption and network status, to assist the network operation and maintenance of staff.</p> <p>3. Remote guidance of experts. Through the camera of AR glasses, experts get the situation of the work site and provide remote guidance.</p> <p>The steps in this use case are as below:</p> <p>1. Using AR glasses (and other external sensors), collect data about the environment, which includes equipment label, port, electricity consumption etc. This step may include barcode / QR code recognition, OCR, Device port recognition etc.</p> <p>2. AI based cognition analysis, perception visualization and other analysis algorithms are applied on the collected data to create a virtual model of planning and design to real environment to assist network designers. This step may include application of Image recognition and tracking, 3D object recognition and tracking, Visual SLAM.</p> <p>3. This model is then used in conjunction with real data for maintenance and optimization by intelligence maintenance assistance system. This step may involve query of the virtual model, analysis of real alarms, cell data, along with the virtual model, to create intelligence assistance for frontline workers. This step may use network data management, system management and core algorithms for the whole system. The output from this step may include 3D models which can be rendered in AR glasses, AI processed network information for display, real-time remote guidance information.</p> <p>4. As the network services evolve and new network functions are plugged in (virtual or physical), the following evolution steps are applied:</p> <ol style="list-style-type: none"> AR app is updated to collect new data, including new equipment data, and new sensors and new environment information. Backstage support system is updated with new data management systems, core algorithms etc <p>5. Periodic or asynchronous reports are produced for human consumption regarding the operation of intelligent maintenance assistant system.</p> <p>6. A software development kit (SDK) may be exposed to 3rd party developers who may develop new applications to analyse the AR-collected data. This may in turn help operators to provide new value-added applications in the intelligent maintenance assistant system.</p>
<p><i>Open issues (as seen by the proponent)</i></p>	<p>1. How autonomy can be applied to simply this process or enhance to enable more?</p> <p>2. The bidirectional relationship: how the (real world) AR+AI system feeds the (digital world) network management & ops (databases) + how the network</p>

	supports the AI+AR system, e.g., by providing inventory and processes information and access to controls?
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	High – Has the potential to impact future service development and evolution.
<i>Reference</i>	

7.29.1 Use case requirements

Critical requirements

- AN-UC029-REQ-001: It is critical that autonomous networks (AN) enable collection of environment data related to network operation and maintenance using automated techniques such as augmented reality (AR) glasses.
- AN-UC029-REQ-002: It is critical that autonomous networks (AN) enable analysis of environment data related to network operation and maintenance using cloud and AI techniques.
- AN-UC029-REQ-003: It is critical that autonomous networks (AN) provide intelligent assistance, rendered using automated techniques such as AR, for network operation and maintenance.

NOTE – The intelligent assistance may be produced using analysis by AI/ML on the collected data from AR.

- AN-UC029-REQ-004: It is critical that autonomous networks (AN) update the data collection mechanisms and data analysis mechanisms along with the result rendering mechanisms based on the analysis by AI/ML on the collected data from AR and the evolution of the underlay networks.
- AN-UC029-REQ-005: It is critical that autonomous networks (AN) provide periodic and/or asynchronous updates to humans about the operation of the intelligent assistant system.

Expected requirements

- AN-UC029-REQ-006: It is expected that autonomous networks (AN) enable exposure of programming capabilities to 3rd party developers for creation of novel applications which can help automated operation and maintenance of network, including evolution and adaptation of network functions.

NOTE – Such novel applications may analyse the data collected using AR, suggest new data collection mechanisms based on gaps in collected data, suggest new analytical methods, or suggest new targets for application of analysis.

7.29.2 Use case specific figures

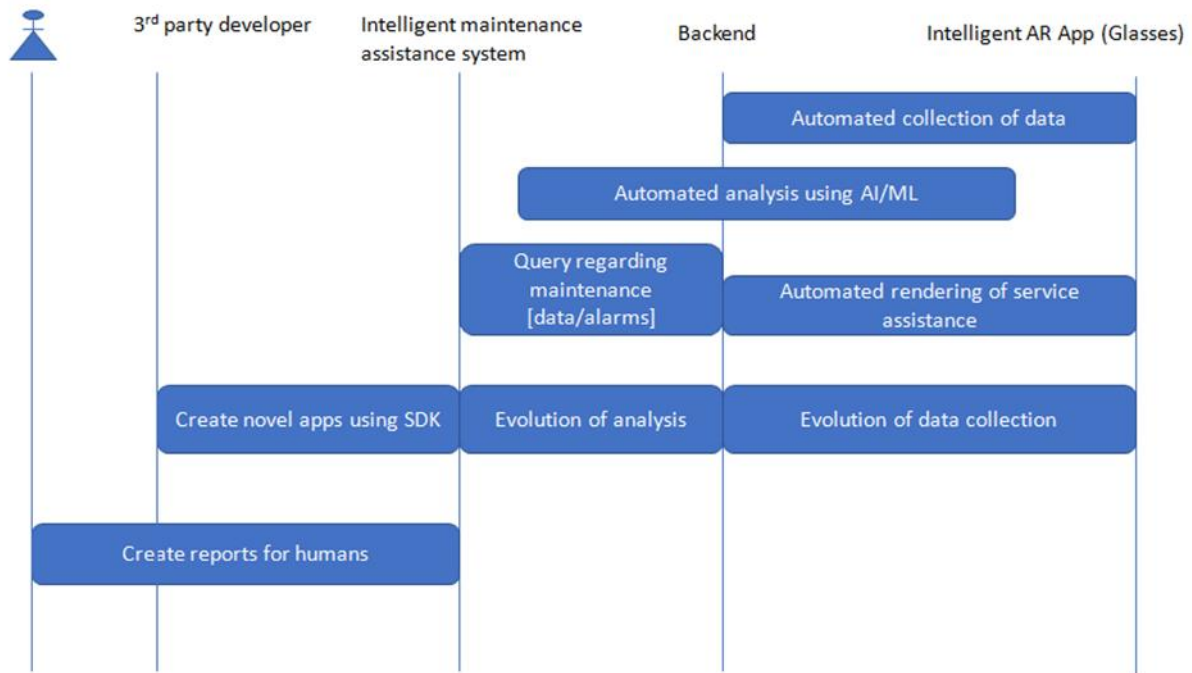


Figure 17 – Actor interaction for intelligent maintenance

7.30 Demand forecasting and live service migration methods in edge computing systems

<i>Use case id</i>	FG-AN-usecase-030
<i>Use case name</i>	Demand forecasting and live service migration methods in edge computing systems
<i>Base contribution</i>	FGAN-I-109
<i>Creation date</i>	01 July 2021
<i>Use case context</i>	Discussions in the weekly meeting during presentation of FGAN-I-109
<i>Use case description</i>	<p>Virtualization and cloudification of services have enabled automation, flexible placement and programmability to network topology. Efficiency of service delivery can be significantly improved using these techniques. However, there are significant challenges to host Ultra-reliable low-latency communication (URLLC) and massive machine type communications (mMTC) services in 5G, in centralized topologies. Monitoring of networks by telco operators have revealed that network topology is not static and load is not uniform over a long service time.</p> <p>This use case describes a dynamic network topology and service placement using the Genetic Algorithm to analyse and predict services. In addition, an efficient forecasting and live migration methods of service as an application to edge computing systems are introduced. This approach can enable intelligent allocation of operator equipment resources, for providing flexible and efficient topologies. Simulation based analysis of results proved that the network equipment efficiency can significantly be increased by these techniques.</p> <p>The optimization of mobile edge computing network performance for a service by addressing the service placement problem is described below:</p> <p>"Match-making" and analytics service is hosted by REX platform, exposed as APIs to 3rd party service providers. Services (e.g., gaming) can now utilize the resources at the edge efficiently. Edge network reports the resource status (and other metadata) to the REX platform via the "REX client"@the edge. 3rd party service provider interfaces with the REX platform via "REX client"@service provider for service deployment at the edge.</p>

	<p>In addition, the following extensions are proposed:</p> <ul style="list-style-type: none"> – REx platform will host "composition" service for controllers (closed loops). – Edge networks will expose APIs to deploy and manage controllers – REx platform will analyse the requirements from applications and manage the composition service towards the edge networks. – The sub/pub mechanism described in [FGAN-I-109_att] is extended to include controller metadata. <ul style="list-style-type: none"> 1. Registration of MEC on the platform will include controller capabilities. 2. Pub MEC status information to the platform will include controller status. 3. Service provider subscribed to this platform receives information about available MECs along with the controllers. <ol style="list-style-type: none"> 1. Connection, Discovery and capability exchange (info exchange) between the hosted (prediction) service and clients (e.g., REx server and client) on the network operator side (edge) and the (application) service providers on the ISP side. 2. Data (including traffic characteristics and controller metadata) is measured and analysed to predict the resource utilization and automation at the edge. 3. Placement of services, migration, and composition of corresponding controllers are managed by REx platform.
<i>Open issues (as seen by the proponent)</i>	<p><i>#Where is the prediction done? Is it done at edge Rex client or Rex server?</i></p> <p><i>#Is the prediction online?real-time?</i></p> <p><i>#Is the use of the intelligent aspects (genetic algorithm) trained/evolved/updated online as the system runs, or is there an a priori training step.</i></p> <p><i>#In slide 8: can we show where is the prediction done?</i></p> <p><i>#Where is the migration decision taken?</i></p> <p><i>#Slide-6: can we map the arrow numbers to the step numbers in the pseudo code in slide 7?</i></p> <p><i>#Investigate service composition.</i></p> <p><i>#Investigate the trust aspects – how to relate/fit.</i></p> <p><i>#Investigate autonomy metrics – to describe the achievement of the use case. in terms of sec, availability – workflow execution vs. adaptability.</i></p>
<i>Notes on use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Notes on priority of the use case</i>	High <ul style="list-style-type: none"> – Has the potential to impact future service development and evolution.
<i>Reference</i>	[b-AI4Good-4]

7.30.1 Use case requirements

Critical requirements

- AN-UC030-REQ-001: It is critical that AN integrate with Edge networks that will expose APIs to deploy and manage controllers.
- AN-UC030-REQ-002: It is critical that AN enable analysis of the requirements from applications and manage the composition service towards the edge networks.
- AN-UC030-REQ-003: It is critical that AN enable sub/pub mechanism including controller metadata.

NOTE – Thus, Registration of MEC on the platform will include controller capabilities. Pub MEC status information to the platform will include controller status. Service provider subscribed to this platform receives information about available MECs along with the controllers.

- AN-UC030-REQ-004: It is critical that AN enable Connection, Discovery and capability exchange (info exchange) between the hosted (prediction) service and clients on the network operator side (edge) and the (application) service providers on the ISP side.
- AN-UC030-REQ-005: It is critical that Placement of services, migration, and composition of corresponding controllers are managed by domain specific orchestrators.

7.30.2 Use case specific figures

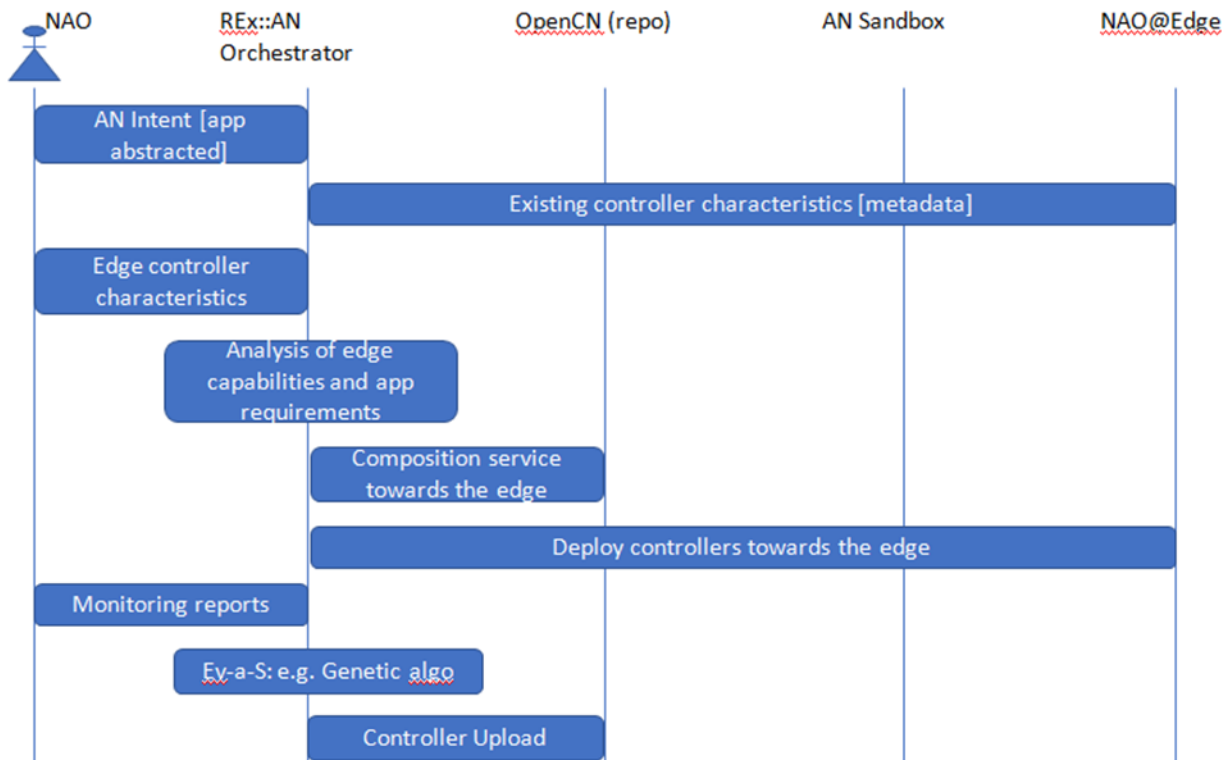


Figure 18 – Actor interaction for Demand forecasting and live service migration

7.31 OpenCN: An open repository of intents for controllers and modules

<i>Use case id</i>	FG-AN-usecase-031
<i>Use case name</i>	OpenCN: An open repository of intents for controllers and modules
<i>Base contribution</i>	None
<i>Creation date</i>	21 July 2021
<i>Use case context</i>	Discussions during 3 rd virtual meeting of FG AN, [ITU-T Y.3176], Webinars supporting ITU AI/ML in 5G Challenge
<i>Use case description</i>	As controllers/closed loops evolve to solve practical problems in the networks, this use case aims to provide a baseline repository (called OpenCN) of intents for different forms of controllers. As in the case of various opensource repositories and AI/ML marketplaces, an open repository will form a baseline for reusable definitions for controllers, provide components for composing and chaining together controllers. In addition, open repo will increase trust in controllers. Metadata related to controllers which describes the controllers and related modules would be enables discovery and other related services like subscription/publication of new controllers etc. Metadata also allows to specify guidelines for integrating controllers with service-x and underlays. Not only does the baseline intents allow stakeholders to reuse, extend and interoperate controller implementations, but it also allows the development of an

	<p>ecosystem of services around it – providing customizations (adaptations), integrations (post experimentations) and finally evolutions.</p> <p>NOTE – Collaboration/integration of OpenCN with closed loop frameworks e.g., ETSI ZSM may be explored.</p> <p>Steps in the use case are as follows:</p> <ul style="list-style-type: none"> – Initial version of controllers are formed from intent or composition from modules (by evolution controllers). – These may be stored in the repo labelled as "untested" or candidate controllers. – Experimentation (Ex) manager pulls the candidates from the repo and (uses AN sandbox to) evaluate and test and compare the controllers. – Evolution (Ev) manager uses the open repo to pull and apply ev strategies. – Operational (Op) controllers are stored in the open repo and pulled and deployed in underlays by various closed loop automation frameworks. <p>NOTE – Standardized intent formats may be used for storing controllers.</p>
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> – There is no unified standard format for intents for closed loops. – There are no opensource solutions to convert intents to closed loops.
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	<p>High</p> <ul style="list-style-type: none"> – Has the potential to impact future service development and evolution.
<i>Reference</i>	

7.31.1 Use case requirements

Critical requirements

- AN-UC031-REQ-001: It is critical that AN enable storage of controllers in an open repository.

NOTE – Experimentation (Ex) manager pulls the candidates from the repo and (uses AN sandbox to) evaluate and test and compare the controllers. Evolution (Ev) manager uses the open repo to pull and apply ev strategies. Operational (Op) controllers are stored in the open repo and pulled and deployed in underlays by various closed loop automation frameworks.

7.31.2 Use case specific figures

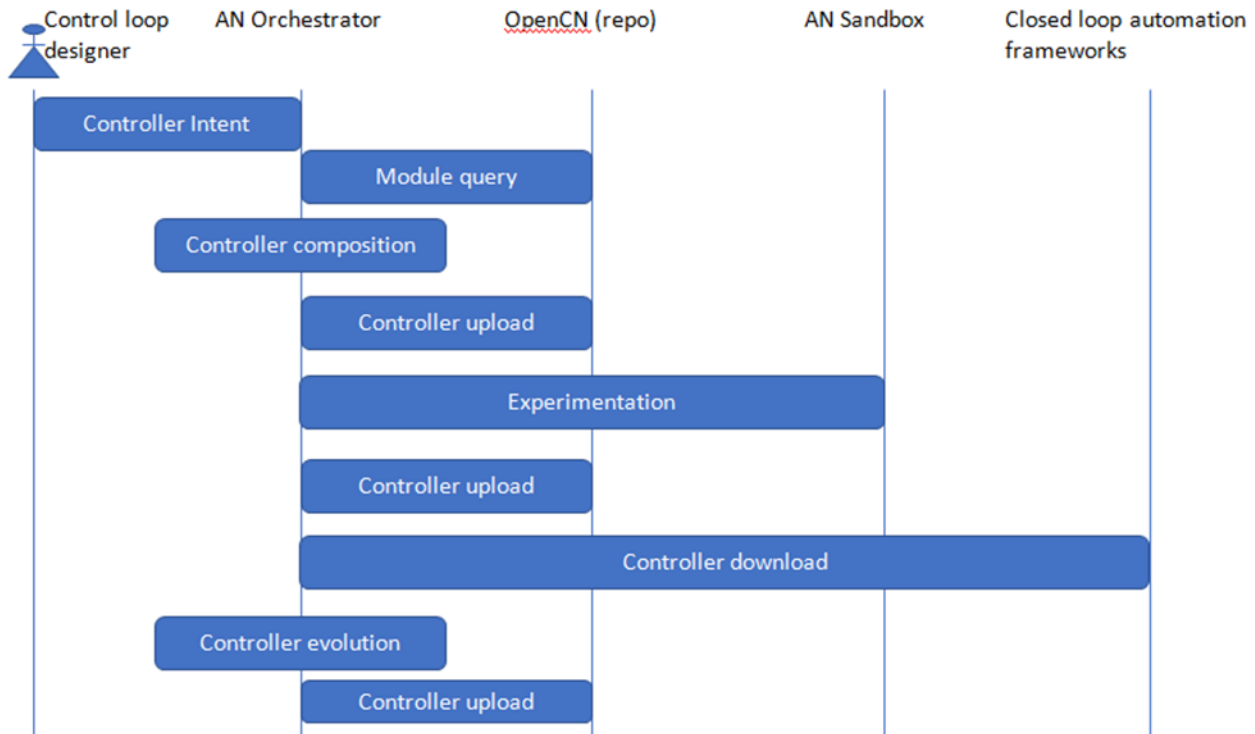


Figure 19 – Actor interaction for open repository of intents

7.32 AI enabled game theory-based mechanism for resource allocation

<i>Use case id</i>	FG-AN-usecase-032
<i>Use case name</i>	AI enabled Game theory based mechanism for resource allocation
<i>Base contribution</i>	FGAN-I-134
<i>Creation date</i>	06 July 2021
<i>Use case context</i>	Discussions during 3 rd virtual meeting of ITU FG AN
<i>Use case description</i>	<p>With the advent of Internet of Things, large number of devices may be trying to connect to the network. Macro base station may be capacity limited to serve these devices. An alternative is device to device communication, but complexity of algorithms and privacy concerns are limiting factors here. Another option is deploying small base stations known as small cells and each small cell try to serve set of users.</p> <p>Microcell, metro cell, Pico cell and femtocells are collectively known as small cells and network containing all these base stations are termed as Heterogeneous networks. To optimize the use of bandwidth, we assume co-channel deployment mechanisms, macro cell and small cells will share the same set of sub-channels. Transmission among small cells may cause interference not only among them but also to the macro cell users. Hence we need to consider efficient resource allocation algorithms to mitigate interference among them so as to satisfy QoS constraints of all the devices/users.</p> <p>Distributed algorithms may be needed to mitigate the interference since there may not be coordination (e.g., X2 interface) among the small cells.</p> <p>In this scenario, we would like to propose distributed resource allocation (subchannel allocation and power allocation to each subchannel) so as to satisfy QoS requirements of all the devices.</p> <p>NOTE – Game theory based model is proposed and to reach equilibrium points we will be proposing both traditional based algorithms like ML algorithms.</p>

Story-1 for the use case: In an enterprise deployment of small cell, say in a factory floor, low latency is required for certain UEs, e.g., robotic arms or self-guided vehicles such as in Industry 4.0 use cases. Macro cell coverage is limited in such indoor factory floors and interference is a problem. Resource allocation for low-latency UEs in the presence of interference and low coverage by macro is the problem addressed in this "story".

- Co-existence of UEs that require low latency service and that does not require low latency service
 - o Each small cell can allocate certain channels (power has to be allocated via game theory) to UEs that require low latency (high priority basis) and for other UEs it can allocate channel and power by playing game model
 - o Power allocation is based on UEs rate requirement
- Serving UEs that all require low latency
 - o The small cell that serves these users can act as a leader and other cells can act as a follower.
 - o Here priority will go to the leader and after allocation of resource by this cell, all others will allocate resources accordingly by playing stackelberg game model.
- Given the measurements such as channel gains observed by user on a particular subchannel along with Interference plus noise on that channel, we need to allocate channel and power per subchannel so as to satisfy user requirements (e.g., rate, low latency)
 - o Here channel allocation matrix and power per subchannel (Known as Nash Equilibrium-NE point) has to be computed iteratively via game theory approach. We can use three different kind of algorithms such as traditional optimised algorithm, reinforcement learning algorithm and genetic algorithm.
 - o We can compare the above algorithms for better usage.
 - o Once we create data set that consists of various measurements-channel gains, interference plus noise, set of sub channels, user per cell, user requirements along with NE point → we can train the supervised learning model and this model sits at the Small BSs to obtain subchannel allocation and power per subchannel depends on the local measurements along with other inputs at each small BSs.
 - o #compare the output from SL with traditional methods.
- Suppose there are multiple users try to access the channel at a particular time and resources are scarce, then we need to schedule the users into different frame durations (msec). For this we consider auctioning mechanism to schedule the user in a particular time slot. Utility for each auction is a function of user requirements such as rate, latency etc.,
- Once we perform user scheduling, then we can allocate channel and power per channel according to algorithm described above.
- In the above procedure we are performing both time domain and frequency domain scheduling separately.

Story-2: surveillance videos – high UL rate is required for cameras – macro cell coverage and interference is problem. Allocation for rate-required UEs in the presence of interference and low coverage by macro. Priority allocation for UL intensive UEs to achieve QoS.

Story-3: power constrained – wearables etc – privacy – user specific data cannot be exposed to 3rd party – to do analytics, no data should be taken out of the trust zone (enterprise or private network). All AI/ML, analytics etc needs to be done within the private network.

	<p>The steps in the use case (related to autonomy) are:</p> <ol style="list-style-type: none"> 1. Controllers are formed based on, e.g., Intents and/or Evolution (Ev) to optimize resource allocation with various considerations including latency, throughput or privacy preserving analytics. 2. Modelling of inter-controller interaction using game theory: "Players" in the game would be the equivalence classes of controllers. "players" may be selected from the evolvable population. 3. (Initial) Strategies/gains/payoffs are defined and initialized for each equivalence class and game is modeled. 4. Players [controllers] can be cooperating or non-cooperating. They may participate in the game based on trust. 5. Modelling of strategies/gains/payoff for each player (controller) may change based on Evolution. 6. Use experimentation (Ex) to study the model (strategies/gains) which is evolved and the elements of trust in this game can be studied. And also to investigate the strategies for maximizing the gains, either by cooperation or non-cooperation. 7. Adaptations to be applied to the Controllers are arrived at – in the form of changes to strategies, gains/payoffs and trust. Formation of new "players" may be part of adaptation. 8. Outer loop: Collect the data from the set of solutions, train the AI/ML model. infer the equilibrium from the new input data using trained model.
<i>Open issues (as seen by the proponent)</i>	<ol style="list-style-type: none"> 1. What are the controllers trying to achieve as players? – use case story. 2. How to quantify computational cost? 3. How to evaluate whether other controllers (players) are trustable (among themselves, whereas so far FG have thought of whether controllers are trusted/deployable)? – take this aspect of trust an input from another layer. <ol style="list-style-type: none"> a. #using AI/ML to overcome some of the disadv. of game theory. What are those? – "the different Nash equilibria and the approaches to reach them could be learnt over some experiments using AI/ML. And then the learnings could be used to optimize these experiments". 4. In the context of trust: players can be selfish, transparent. This understanding may change as Ev progress. Trust index can be studied. – "this is a new problem". 5. How to use RL (need quick feedback) here? use real network or simulated data? – Use different RL mechanisms like 0 regret. Use simulated data. #convergence may take long time in RL. Use DTwin to introduce RL in AN. 6. Small cell coverage is small, num of users is less, so the distribution of service requirement is sparse (?). – as the coverage density increases, to serve large num of users... capacity also increases...specific indoor scenario like factory...but for the resource allocation may be more complicated for macro cell, interference with macro may be considered, game with macro (coordination with macro) may be considered later.
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	High Justification: the architecture components derived and introduced here will help in defining meta-evolution controllers [I-98].
<i>Reference</i>	[b-Sankar-1, [b-Sankar-2], [b-Ahmad], [b-Al-Turjman], [b-Ciavaglia-2]

7.32.1 Use case requirements

Critical requirements

- AN-UC032-REQ-001: It is critical that autonomous networks (AN) enable characterisation of controllers using metadata, which may be updated dynamically based on monitoring of the controllers.

NOTE – The metadata associated with controllers would be used for modelling controllers as players. Example – the closed loops aiming to optimize transmit power and those intending to optimize coverage may be modelled as players in a game.

Expected requirements

- AN-UC032-REQ-002: It is expected that autonomous networks (AN) enable experimentation with various gaming strategies, payoffs and equilibria with controllers as players.

NOTE – Experimentation may be conducted in the Sandbox and coordinated e.g., by experimentation manager, may result in analysis of strategies, payoffs and equilibria.

- AN-UC032-REQ-003: It is expected that autonomous networks (AN) enable classification of controllers with respect to trustability of controllers.

NOTE – Parameters related to measurement of trust as applied to controllers and the methods of classification may be out of scope of this particular use case.

- AN-UC032-REQ-004: It is expected that autonomous networks (AN) enable analysis of experimentation results with AI/ML based techniques.

NOTE – Learnings from AI/ML may be used in optimizing the gaming strategies, payoffs and equilibria.

- AN-UC032-REQ-005: It is expected that autonomous networks (AN) enable adaptation of controllers with new strategies.

NOTE – Learnings from AI/ML may be used in optimizing the gaming strategies, payoffs and equilibria.

- AN-UC032-REQ-006: It is expected that autonomous networks (AN) enable derivation and application of different combinations of game theory mechanisms such as auction theory based on the use case specification.

NOTE – Use case specification may be captured and formalized in the form of Intent. Derivation of game theory mechanisms such as auctioneer may use human or automated mechanisms. Application or integration of such mechanisms in underlays may use specific architecture and interface considerations such as in Open RAN.

7.32.2 Use case specific figure

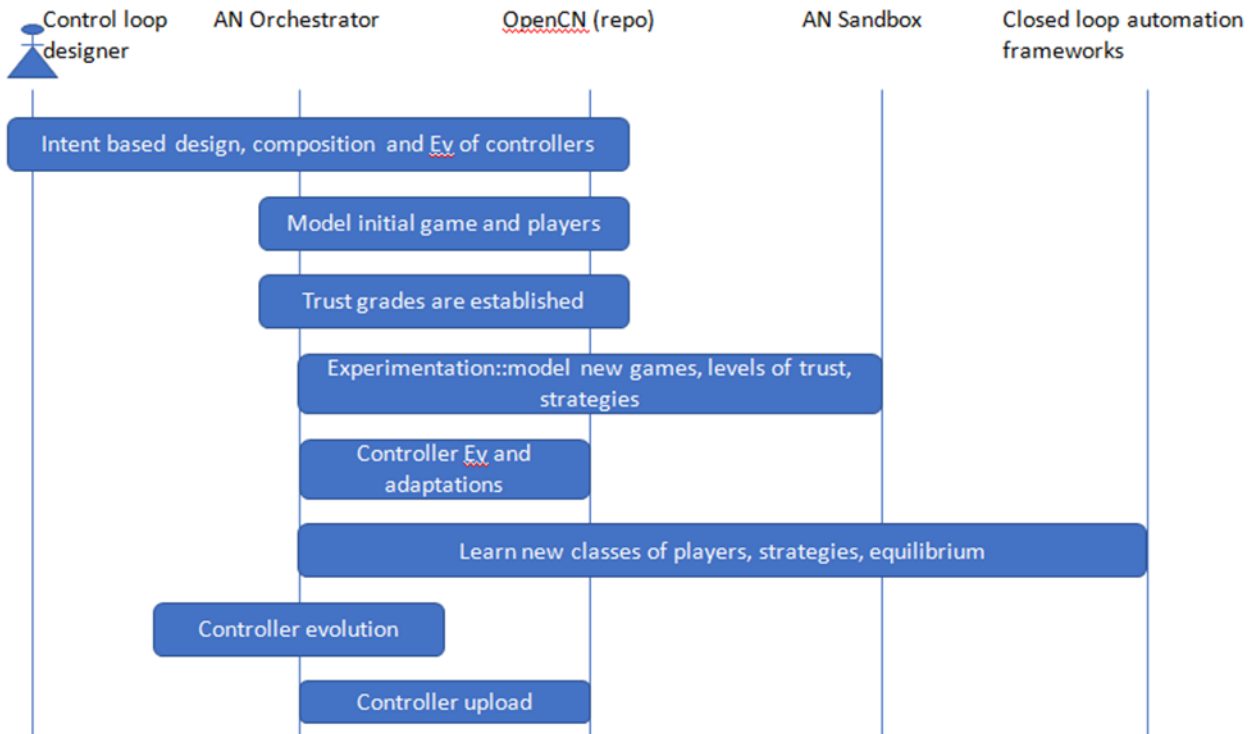
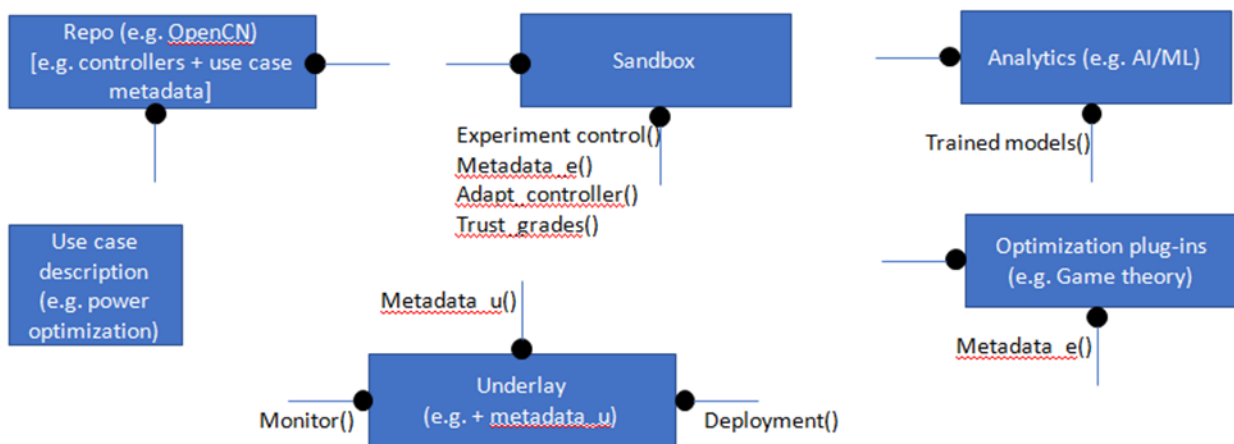


Figure 20 – AI enabled Game theory based resource allocation



Metadata_e() = metadata for experiments e.g. strategies, payoffs
Metadata_u() = metadata characteristics of underlays e.g. Inputs for mapping abstract controllers to underlay specific instances
 Use case metadata = metadata which describes the use case which captures the constraints and helps in mapping controllers to deployable instances

Figure 21 – Component cloud for AI enabled Game theory based resource allocation

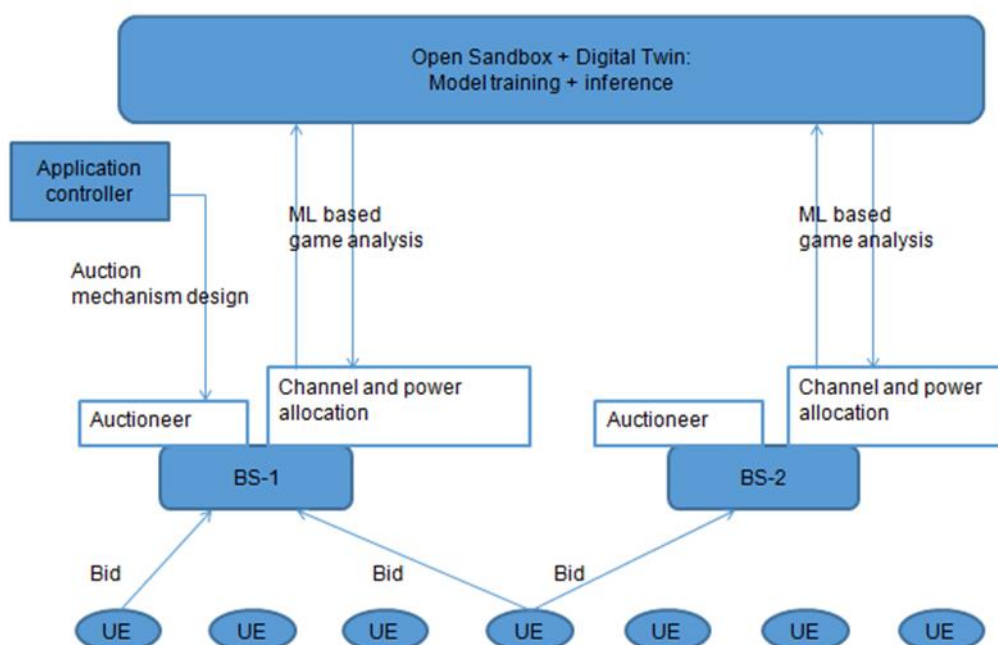


Figure 22 – AI enabled Game theory based resource allocation

7.33 Service automation using workflows

<i>Use case id</i>	FG-AN-usecase-033
<i>Use case name</i>	Service automation using workflows
<i>Base contribution</i>	FGAN-I-135
<i>Creation date</i>	27 July 2021
<i>Use case context</i>	Follow up discussions after TIP webinar on automation
<i>Use case description</i>	<p>Network automation scenarios in future networks include large scale automation of management of network devices, services and retrieval of operational state data from a network. User specific workflows, along with modularized tasks are one of the mechanisms to achieve this automation.</p> <p>Combined with an interworking of OpenConfig NETCONF & YANG models, vendor native models, and the CLI, use specific, customized solutions can be created and dockerized containers can be designed and tested.</p> <p>The workflows are defined using a JSON based domain specific language (DSL) by wiring a set of tasks together. The tasks are either control tasks (fork, conditional, etc.) or application tasks (i.e., encoding a file) that are executed on a remote device. Atomic tasks are chained together into more complex workflows.</p> <p>NOTE – The FRINX Machine distribution comes pre-loaded with a number of standardized workflows.</p> <p><u>Steps in this use case are as follows:</u></p> <p>Step-1: create or compose workflow:</p> <p>Operations or functions on workflows include: In addition to create, the workflow designer can also:</p> <ul style="list-style-type: none"> – edit a workflow; – delete a workflow. <p>These operations may be achieved using a workflow manager.</p> <p>This may include an API based interface or a GUI based interface.</p> <p>A graphical user interface may be used to create, edit or run workflows and monitor any open tasks. The GUI may also help in explainability.</p>

	<p>NOTE – This would help to on-board new services (e.g., in network underlays) and view their status.</p> <p>NOTE – The composing step (from the GUI) may produce output in a generic form (e.g., TOSCA or YANG) and translating this representation of workflows into deployable instances can be using a generic and can take different types of inputs and produce different types outputs.</p> <p>Step-2: store in Resource database: Workflow specification and execution data are stored in a resource database.</p> <p>Step-3: link Tasks: A task corresponds to a worker utilized in the workflow. Tasks in our workflow may receive input parameters and execution logic of our task may be implemented in python functions called worker. Worker tasks may be registered in the main python file "main.py" in the same directory where you just created your worker.</p> <p>NOTE – All workers which one want to use in Frinx Machine must be included in this file.</p> <p>Step-4: monitor: The following service states may be mapped to workflows:</p> <ul style="list-style-type: none"> – Experimental state: workflows which are deployed in the sandbox are in experimental state. In combination with simulators, these are tested and experimented upon. – Evolutionary state: workflows which are in ev state are selected for evolution, and based on ev strategies, various experiments may be designed for them. – Deployed state: workflows which are in deployed state are in combination with service-x, acting upon real underlays. They may be monitored for performance and other parameters. <p>Step-5: deploy workflows: Workflows may be deployed on simulated underlays and their performance and benchmarking may be tested and monitored. Workflows may also be deployed on real network underlays once their performance in the experimentation is satisfactory.</p> <p>Description of the relation (if any) of the use case with autonomous behaviour or the key technical enablers.</p> <ul style="list-style-type: none"> – In the context of the use case, controllers (closed loops) are represented as workflows. Modules are modelled as tasks. – Controller specification and module specifications are created using designer, sanity checked and stored in the resource db. – Workflow manager is used to visualize the controllers, monitor and analyse. – Deploy will link service-x with controllers.
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> • FFS
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	High Justification: gives potential collaboration opportunities and reuse possibilities.
<i>Reference</i>	[b-FRINX-1], [b-FRINX], [b-TIP 5G]

7.33.1 Use case requirements

- AN-UC033-REQ-001: It is critical that autonomous networks (AN) enable creation of controllers in a generic format agnostic to the type and characteristics of the underlay network.

NOTE – The generic format may not include deployment specific details. This allows design time flexibility and abstraction.

- AN-UC033-REQ-002: It is critical that autonomous networks (AN) enable translation of controller specifications from generic format to include underlay specific details.

NOTE – The translation of controller specifications may include, for example, steps like replacing and augmenting abstracted parameters with underlay specific parameters and such customizations.

- AN-UC033-REQ-003: It is critical that autonomous networks (AN) enable storage of generic and underlay-specific representation of controllers in a repository.

Expected requirements

- AN-UC033-REQ-004: It is expected that autonomous networks (AN) enable different implementations of translation from generic to underlay-specific representation of controllers.

NOTE – Workflow managers which implement the translation may be provided by different vendors.

- AN-UC033-REQ-005: It is expected that autonomous networks (AN) enable storage of different classes of controllers in a repository.

NOTE – Examples of different classes of controllers are (a) initial representation of controllers, (b) those which are experimented with corresponding metadata (results), (c) those which are deployed with corresponding metadata (from monitoring).

7.33.2 Use case specific figures

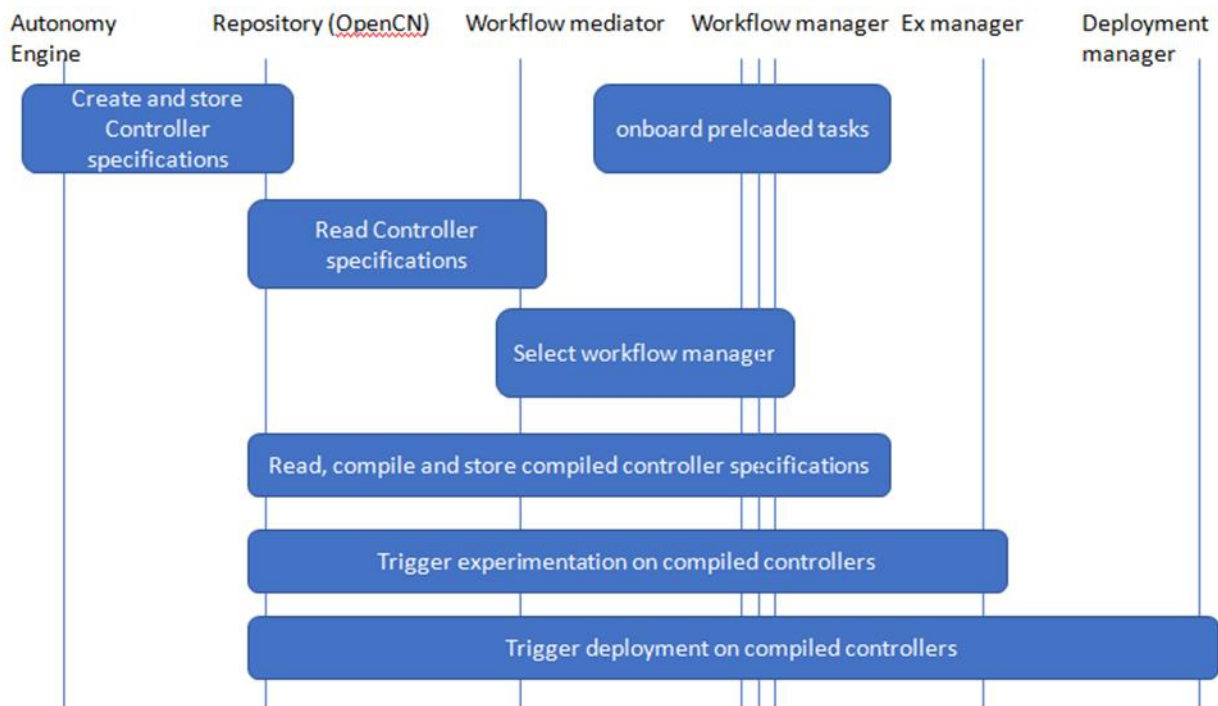


Figure 23 – Actor interaction for Service automation using workflows

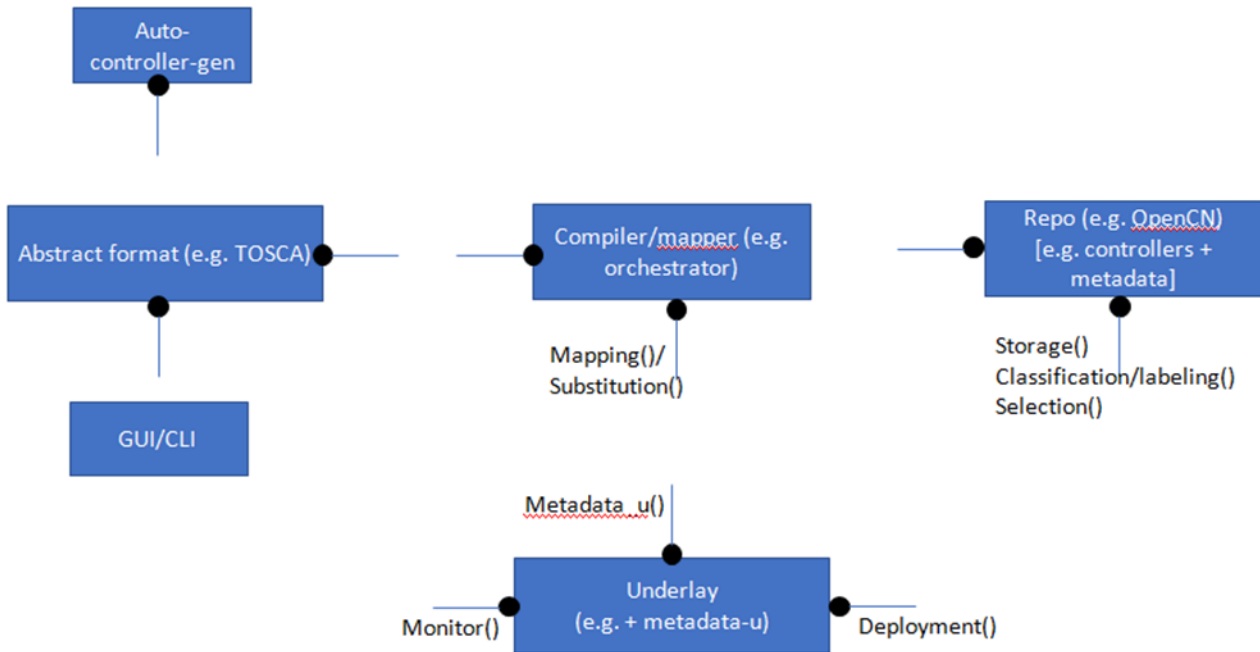


Figure 24 – Component cloud for Service automation using workflows

7.34 Disaggregation and placement of in-network programs

<i>Use case id</i>	FG-AN-usecase-034
<i>Use case name</i>	Disaggregation and Placement of In-Network Programs
<i>Base contribution</i>	[FGAN-I-124]
<i>Creation date</i>	08 Sep 2021
<i>Use case context</i>	Follow up discussions after [FGAN-I-124]
<i>Use case description</i>	<p>Concepts and tools based on the overarching vision of Software-Defined Networking such as host based (packet processing), Middleboxes, NFV (Network Function Virtualization), OpenFlow and configurable flow tables, and Programmable switch ASICs are well known. Support for P4 Language consortium has grown over time and it has been central to enabling dataplane programmability. In-Network Programs such as Forward Error Correction (FEC), traffic compression, encapsulation/decapsulation, etc are typical applications in dataplanes. However, the current programming paradigms map dataplane programs 1:1 to devices and resource dedicated to the program, executing on a single target, limiting to the scope of "programmability of the network" and hence creating a mismatch with the overarching vision of SDN.</p> <p>Decomposing of dataplane programs into a suitable mix of dataplanes is needed.</p> <p>Steps in this use case are as follows:</p> <p>A. "loosely coupled" integration of flightplan with AN:</p> <p>Step-1: Given a controller specification, selection of "in-network controllers" e.g., Dataplane programs, is made by the selection controller.</p> <p>Step-2: Flightplan helps in segmenting, planning and allocation/mapping to devices.</p> <p>Step-3: Other forms of "underlays" e.g., FRINX Machine, ZSM managed domains, would host their own ways of achieving the above-mentioned integration of controllers in their service domains. This forms the "application" (or deployment) side of controller lifecycle.</p>

	<p>B. "tightly coupled" integration of flightplan with AN:</p> <p>Step-4: As mentioned in slide 27 of [FGAN-I-124_att], flightplan may interface and "collaborate" with AN orchestrator (or different controllers in the AN domain) to achieve the following:</p> <ol style="list-style-type: none"> Segmentation of dataplane programs based factors known to AN e.g., resource needs, overall E2E controller deployments across different domains like ZSM or FRINX. Diagnose and debug and analyse various "possibilities" for solutions. Hand-over control between various dataplanes. Placement of controllers. Analysis of performance KPIs. <p>Step-5: "Suggest" or "bubble up" changes or evolutions to controllers based on flightplan's analysis of constraints and KPIs.</p>
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> Currently flightplan does not do step-4 and 5 above. It is TBD how to achieve these. Reference points and interfaces for integration between flightplan and AN are to be studied and TBD.
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	High Justification: gives potential collaboration opportunities and reuse possibilities.
<i>Reference</i>	[b-Flightplan]

7.34.1 Use case requirements

- AN-UC034-REQ-001: It is critical that AN support both loosely coupled and tightly coupled integration mechanisms with underlays.

NOTE 1 – Examples of underlays are workflow management mechanisms like FRINX, closed loop automation frameworks like ETSI ZSM, and in-network dataplane program management mechanisms like Flightplan. Example of loosely coupled integration is passing down intents from AN to Flightplan (which further analyses it to derive program profile). Example of tightly coupled integration is exchange of granular information e.g., current resource allocation status, current KPIs monitored, providing control and visibility over placement of controllers in devices.

NOTE 2 – The loosely coupled integration allows for more autonomy in the underlay.

- AN-UC034-REQ-002: It is critical that underlays support loosely coupled integration mechanisms with AN.
- AN-UC034-REQ-003: It is critical that AN support discovery of capability with respect to level of integration provided by the underlay.
- AN-UC034-REQ-004: It is critical that AN decide the "application" and deployment of controllers to various underlays based on the E2E requirements of the use case.
- AN-UC034-REQ-005: It is expected that underlays support tightly coupled integration mechanisms with AN.

7.34.2 Use case specific figures

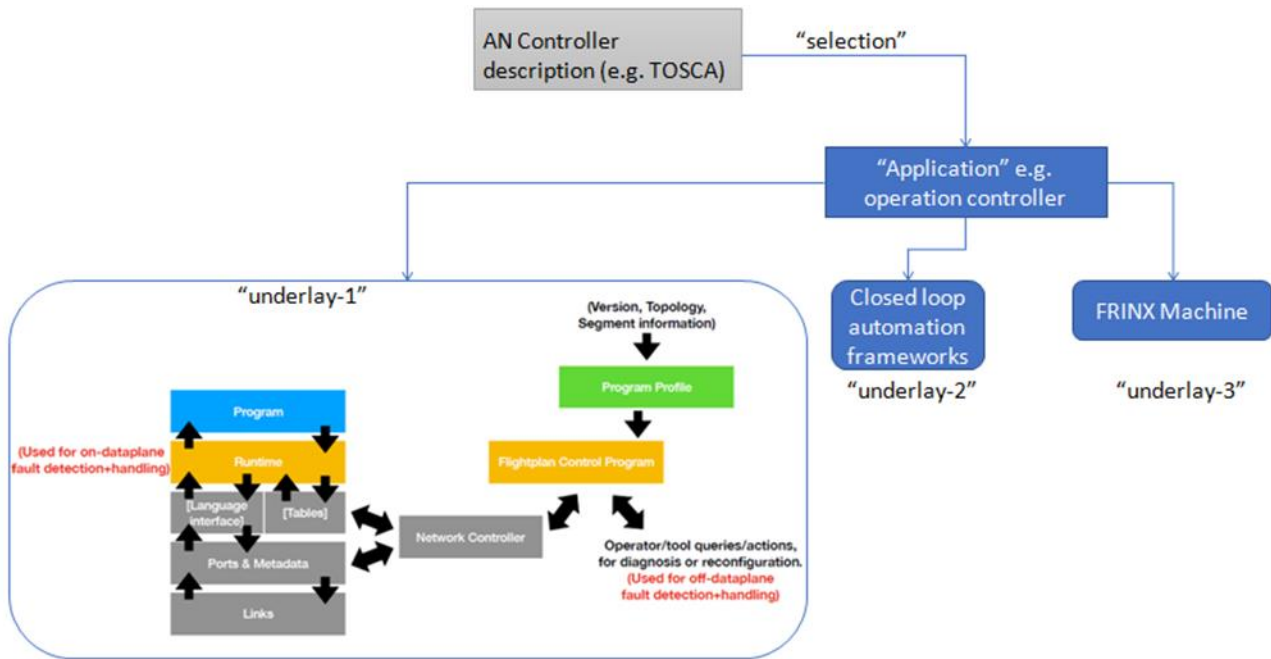


Figure 25 – Disaggregation and Placement of In-Network Programs

7.35 A fast and lightweight autoML library (FLAML)

Use case id	FG-AN-usecase-035
Use case name	FLAML: A Fast and Lightweight AutoML Library
Base contribution	FGAN-I-134
Creation date	16 Sep 2021
Use case context	
Use case description	<p>Currently, selecting learners and hyper parameters for each learner is a tedious and manual task. Fast and Lightweight AutoML (FLAML) is a lightweight Python library that finds accurate machine learning models automatically, efficiently and economically.</p> <p>FLAML leverages the structure of the search space to choose a search order optimized for both cost and error. For example, the system tends to propose cheap configurations at the beginning stage of the search, but quickly moves to configurations with high model complexity and large sample size when needed in the later stage of the search.</p> <p>FLAML integrates several simple but effective search strategies into an adaptive system.</p> <p>FLAML can:</p> <ol style="list-style-type: none"> 1. optimize with low latency; 2. can handle large datasets; 3. not restricted to fixed set of configs; 4. easy to add new/custom learners; 5. can cold start. <p>Some of the optimizations done are related to:</p> <ol style="list-style-type: none"> 1. which trials to invoke 2. in what order? <p>Some of the proposed properties of optimizations in FLAML are:</p> <ol style="list-style-type: none"> 1. suitable sample size;

	<ol style="list-style-type: none"> 2. resample; 3. fair chance; 4. optimal trial. <p>Steps related to use case:</p> <ol style="list-style-type: none"> 1. Overall use case for autonomous network is specified in the intent to AN. 2. The requirements for ML use case and hence ML intents are derived from the overall use case. 3. Based on the knowledge base [or human guidance], AN configures FLAML with the ML intent. 4. It is left to FLAML to select the learner, do trials, and adapt/optimize the ML model's parameters. 5. It is left to FLAML to tightly or loosely couple with the AN domain, to utilize its experimentation manager, analytics or Sandbox or knowledge base (KB).
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> • What are the experimentation/trials? where do they come in? • How do you determine the order for trials? and what detail you need from the domain e.g., do we require knowledge from the domain to apply this? if so, where is it captured / used? • How to integrate this with overall architecture? • How to compare FLAML with competitors? • What is the level of autonomy available to FLAML?
<i>Notes on use case category</i>	Cat 1: Describes a scenario related to core autonomous behaviour itself.
<i>Notes on priority of the use case</i>	High Justification: gives potential collaboration opportunities and reuse possibilities.
<i>Reference</i>	[b-Wang-1], [b-FLAML], [b-Microsoft], [b-Wu-1], [b-Wu-2]

7.35.1 Use case requirements

- AN-UC035-REQ-001: It is critical that AN enable capturing of overall use case in an AN intent and derivation of requirements for ML use case and hence ML intents from the overall use case.
- AN-UC035-REQ-002: It is critical that Based on the knowledge base [or human guidance], AN configures ML optimization tools with the ML intent.

NOTE – FLAML is an example of ML optimization tool. It is left to FLAML to select the learner, do trials, and adapt/optimize the ML model's parameters.

- AN-UC035-REQ-003: It is critical that based on the capabilities of the ML optimization tool, AN tightly or loosely couples with the tool, to utilize its experimentation manager, analytics or Sandbox or knowledge base (KB).

7.35.2 Use case specific figures

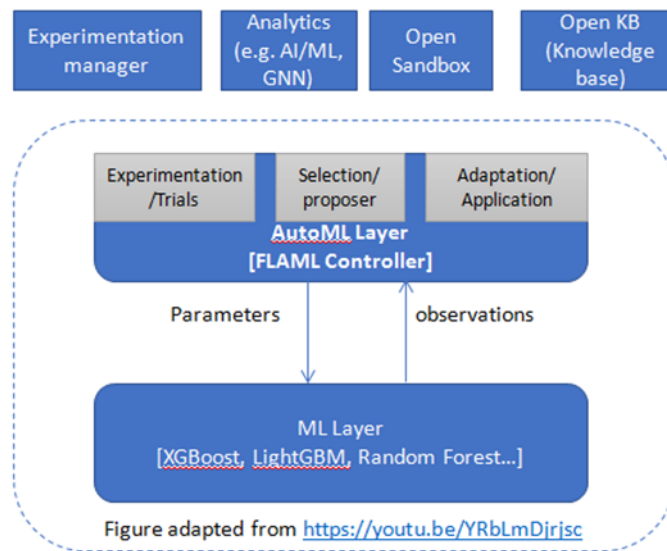


Figure 26 – FLAML Controller for AutoML

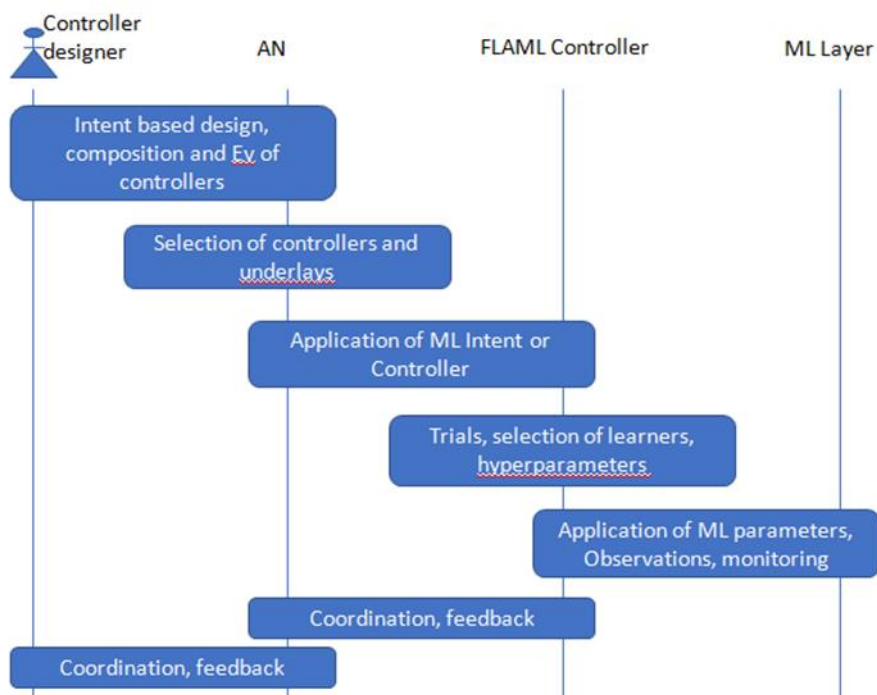


Figure 27 – Actor interaction for FLAML

7.36 Connected AI (CAI) testbed: Testbed for 5G connected artificial intelligence on virtualized networks

Use case id	FG-AN-usecase-036
Use case name	Connected AI (CAI) testbed: Testbed for 5G Connected Artificial Intelligence on Virtualized Networks
Base contribution	FGAN-I-148
Creation date	22 Sep 2021
Use case context	Presentation by LASSE/UFPA on 29 Sep 2021 [adhoc meeting of FG AN]

<i>Use case description</i>	<p>5G testbeds are an important alternative to simulators and many have been recently described, emphasizing aspects such as cloud functionalities, management and orchestration. This use case describes a 5G mobile network testbed with a virtualized and orchestrated structure using containers, which focuses on integration to artificial intelligence (AI) applications.</p> <p>Two use cases are described, one for RAN slicing and another for the placement of VNFs according to application requirements. Focus is on application of AI to RAN and transport network, including fronthaul and backhaul.</p> <p>The SDN and RAN controllers work as information sources about the network. They also work as agents to dynamically change the mobile and the computer network. An AI agent performs different actions in the testbed according to the application, using the information provided by SDN and RAN controllers to train and execute in test stage its neural networks. The ML workloads are orchestrated along the cluster to provide the AI agent processes.</p> <p>Steps related to use case:</p> <ol style="list-style-type: none"> 1. Overall use case for autonomous network is specified in the intent to AN. 2. The requirements for AI Agent and hence ML intents are derived from the overall use case. 3. Based on the knowledge base [or human guidance], AN configures AI Agent with the ML intent. 4. It is left to AI Agent to select the model, and optimize the ML model's parameters. 5. It is left to AI Agent to tightly or loosely couple with the AN domain, to utilize its experimentation manager, analytics or Sandbox or knowledge base (KB).
<i>Open issues (as seen by the proponent)</i>	<ul style="list-style-type: none"> • Mapping to pipeline [ITU-T Y.3172] and relation to FLAML use case.
<i>Notes on use case category</i>	Cat 2
<i>Notes on priority of the use case</i>	<p>High</p> <p>Justification: gives potential collaboration opportunities and reuse possibilities.</p>
<i>Reference</i>	[b-Nahum]

7.36.1 Use case requirements

- AN-UC036-REQ-001: It is critical that Based on the knowledge base [or human guidance], AN configures AI agents.

NOTE – AI agents may in turn select the model and optimize the ML model's parameters in the ML pipeline to be deployed in the underlay.

- AN-UC036-REQ-002: It is critical that based on the capabilities of the AI agent, AN tightly or loosely couples with the AI agent, to utilize its experimentation manager, analytics or Sandbox or knowledge base (KB).

7.36.2 Use case specific figures

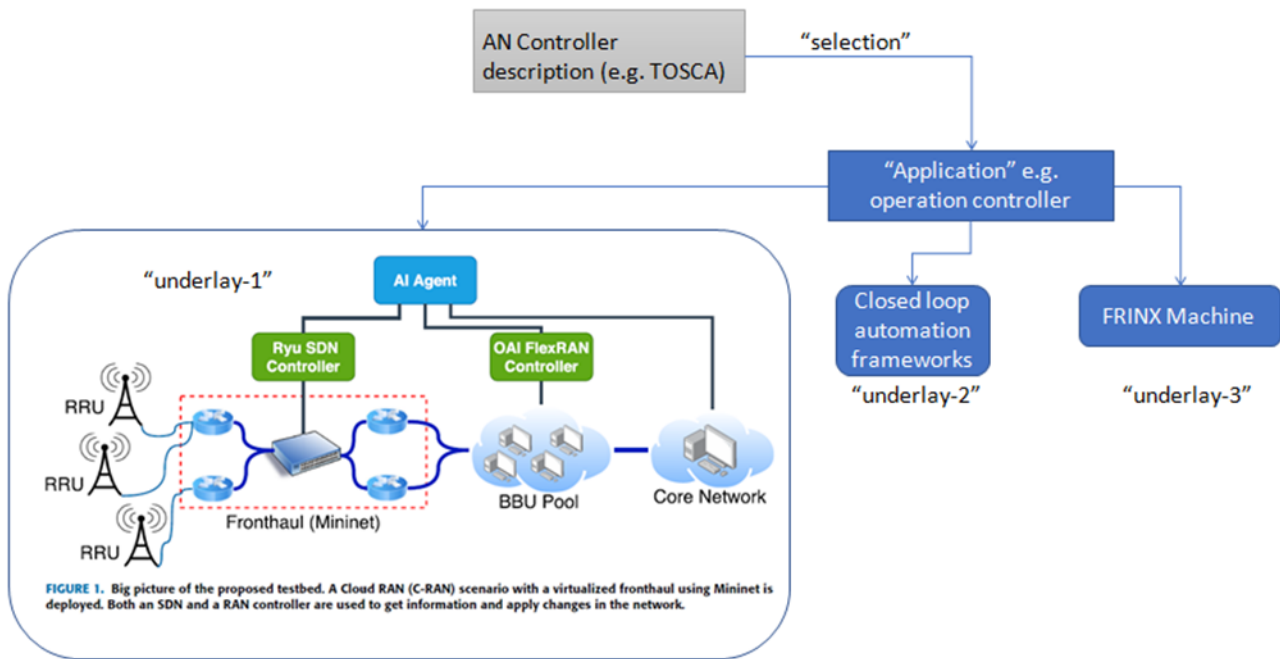


Figure 28 – Connected AI (CAI) testbed

7.37 Negotiated boundaries in AN for seamless network sharing

<i>Use case id</i>	FG-AN-usecase-037
<i>Use case name</i>	Negotiated boundaries in AN for seamless network sharing
<i>Base contribution</i>	
<i>Creation date</i>	28/09/2021
<i>Use case context</i>	Dynamic network sharing in AN
<i>Use case description</i>	<p>We see different networks operated and deployed by different operators. Generally, the networks also operate in shared settings. The operators or interested parties like Mobile Virtual Network Operators (MVNOs) share the network for various reasons like reduce CAPEX and OPEX (reduce TCO in general), providing coverage for the users. There are some underlying arrangements for sharing the network like RAN sharing, spectrum sharing, core network and transport network sharing. These arrangements are governed by the agreement between the interested parties. The arrangements possibly could be network slicing, 3GPP based network sharing or some custom vendor solution. The arrangements are rigid and not flexible because of a prior agreement between the parties. The prior agreement creates a boundary between the operators, and any change in the boundary requires a revisit to the agreement. E.g., the changes are forming new agreements with different operators, withdrawing the existing agreement, and changing service level agreements (SLA).</p> <p>From the AN perspective, the strict boundary is a roadblock towards autonomy. The boundaries shall be flexible, scale in and out dynamically without involving any central authority (here operator/human). The AN decides the scaling of boundaries (in/out) based on the requirement. The control loops, intent or some prediction algorithm can generate requirements for the action of AN. The AN shall adapt and provide the associated management like policing, billing and other configured elements related to the shared arrangement between the interested parties. The AN shall have the capability to negotiate the boundaries through the</p>

	<p>adapted agreements between the interested parties. The negotiation of boundaries means the AN can independently change the agreements in runtime.</p> <p>Consider the following example of load balancing:</p> <ol style="list-style-type: none"> 1. Consider a scenario of base stations deployed by different operators in a single coverage area. 2. The base station is loaded, and there is a need to balance the load for serving the users with the desired quality of service. 3. In the case of non-AN, the base station can balance the load to different operators base stations. However, it can only scale out to the pre-defined operators in the agreement, and those base stations can also be loaded. To scale out to the other base stations, it needs an operator(human) in the loop who then complete the sharing agreement and provision arrangements. 4. In AN case, the base station can identify the requirement to load balance the users to other base stations. The AN acts independently to adapt the agreement and scale its coverage via other base stations with reduced intervention of operator(human). The AN provision the associated agreements dynamically as per the requirement.
<i>Open issues</i>	<ol style="list-style-type: none"> 1. It may require AN to trust other operator's AN <p>Q1. side-effect: standard/interoperable format for derived context: "federated AN". sharing data/derived context, knowledge, insights, generated from AN, may be generalized and applicable across operators if possible. Representation of KB (generalized) may be useful, but practically it may be use case specific. e.g., standard representation of config for BS. //federated learning is just an analogous domain and not a normative detail here.</p>
<i>Use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Reference</i>	

7.37.1 Use case requirements

Critical requirements

- AN-UC037-REQ-001: It is critical that autonomous networks support seamless autonomous behaviour in case of underlay networks that support scaling across shared resource pools and across administrative domains.

NOTE 1 – Examples of seamless autonomous behaviour may include inclusion (or exclusion) of new controllers or closed loops or services or devices into the scaled AN domain.

NOTE 2 – Examples of shared resource pools are CPU cores, memory, RAN resources like spectrum frequencies, or network resources like ports.

NOTE 3 – Examples of administrative domains are base stations, transport network or core network functions which are owned, maintained and operated by different network operators.

Expected requirements

- AN-UC037-REQ-002: It is expected that autonomous networks reuse existing mechanisms for dynamic management of agreements between different administrative domains of network operators to achieve seamless autonomous behaviour in case of underlay networks that support scaling across shared resource pools and across administrative domains.

NOTE – Examples of existing mechanisms for dynamic management of agreements are block-chain mechanisms or smart contracts.

7.37.2 Use case specific figures

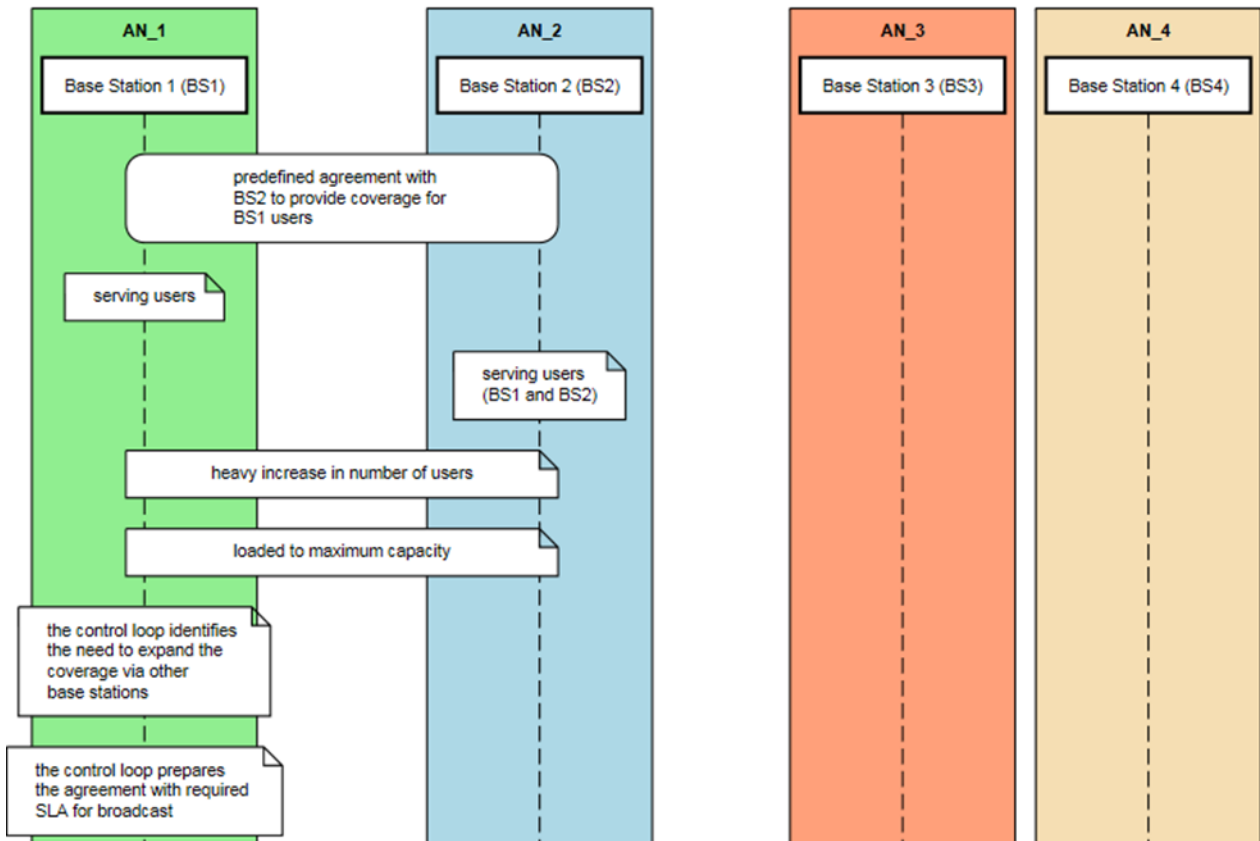


Figure 29 – Actor interaction for Negotiated boundaries in AN for seamless network sharing: part-1

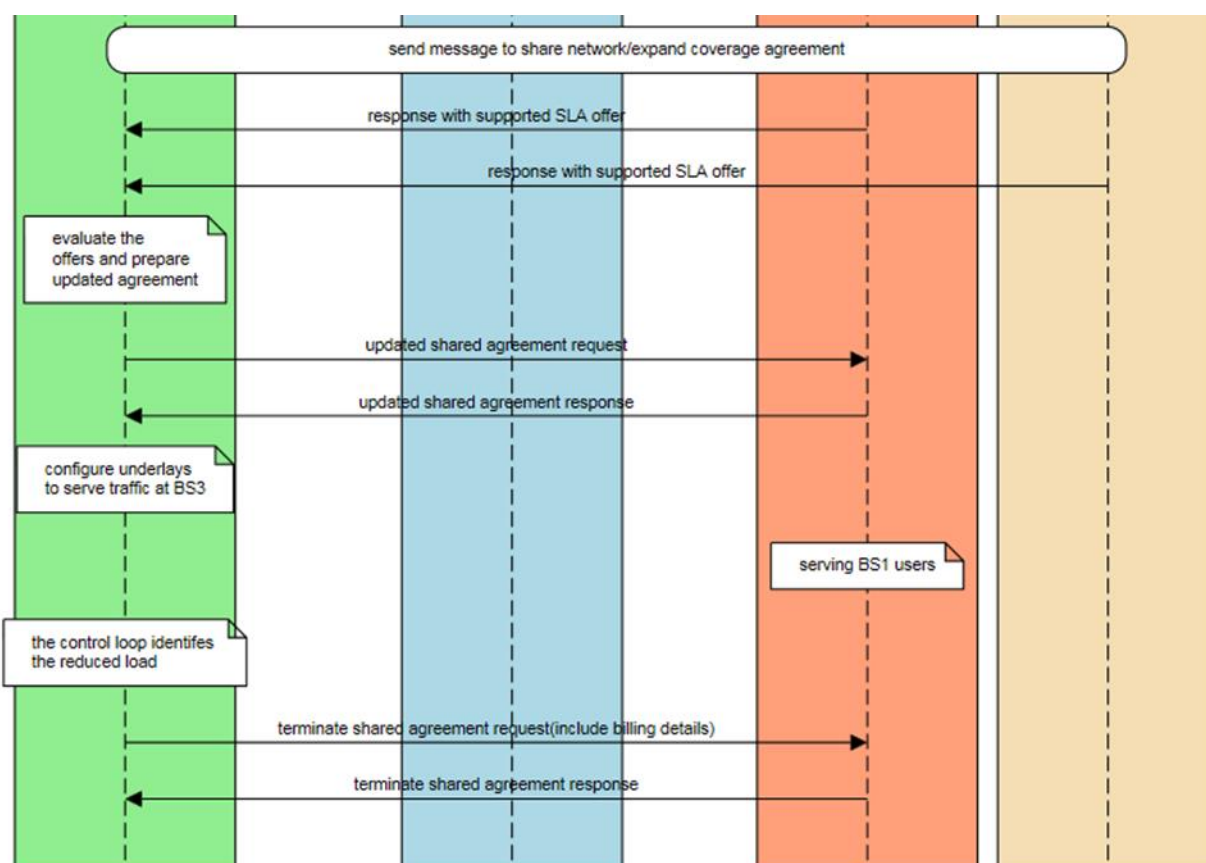


Figure 30 – Actor interaction for Negotiated boundaries in AN for seamless network sharing: part-2

7.38 AN enabled end-to-end supply chain

<i>Use case id</i>	FG-AN-usecase-038
<i>Use case name</i>	AN enabled end-to-end supply chain
<i>Base contribution</i>	
<i>Creation date</i>	28/09/2021
<i>Use case context</i>	Orchestration of supply chain in the AN
<i>Use case description</i>	<p>Today the supply chain for the networks is highly operator dependent. The operator drives the choice of equipment, procurement, testing and deployment. The planning tools can help to automate the deployment process. There are companies deploying drones for inspection and monitoring of deployed base stations [b-Sharma]. Such inspection by drones is a step towards automation and autonomy. The domains like Mission Critical require the on-demand deployment of the network. The operator in the loop for such mission-critical deployment could incur delays for emergency services. The supply chain's end-to-end orchestration final stage is physically deployed network equipment from the supply chain perspective. The deployment means the network equipment is up and running. The network equipment could be virtual or physical. In an ultra-dense deployment world, the scale of equipment in numbers is very large and requires autonomous management. It is very difficult for an operator to manage millions of equipment deployed in different geographies.</p> <p>We see that the end-to-end orchestration of the supply chain needs a revisit in the AN context. There is a need to delegate the control from the operator to AN and remove the dependency of the operator. The AN can act independently and decide the orchestration of the supply chain as per the requirements. We understand that AN cannot do everything like physical things – power-up of equipment, cabling,</p>

	<p>but it can certainly schedule those events, notify required teams, and monitor progress. Furthermore, the AN exposes interfaces for operators or orchestrators to feed the vendor lists, equipment lists for bootstrapping and update process. The AN drives the end-to-end supply chain orchestration with minimal intervention of the operator in run time.</p> <p>Consider the following example:</p> <ol style="list-style-type: none"> 1. The operator feeds the list of vendors or procurement of equipment in AN. It is similar to the source of the image in the docker file. Over a period of time, the AN could learn which equipment from which vendor and rate the vendors based on the performance of their equipment. 2. AN identifies the need to provision network equipment for various reasons like replacement for fault equipment, expansion of the network's capacity, new sites, upgrade, emergency deployment. 3. AN understands the need possibly through some optimization algorithm, direct requirement through intents, other orchestration mechanisms, prediction-based control loops. 4. AN drives the supply chain by procuring, testing the equipment and finally deploying the equipment. The testing requires operator intervention for physical activities. The AN runs the automated test procedures and tallies the results to decide the deployment of equipment. 5. AN configures the underlying equipment and newly deployed equipment for integration to go live.
<i>Open issues</i>	1. Required interfaces to communicate with operator and orchestration mechanisms.
<i>Use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Reference</i>	[b-Sharma]

7.38.1 Use case requirements

Critical requirements

- AN-UC038-REQ-001: It is critical that autonomous networks support interface towards and orchestration of software and hardware inventory management systems in underlays.
- AN-UC038-REQ-002: It is critical that autonomous networks support interface towards respective teams (human or bot) or operations centre of the software and hardware components/equipment/systems.

NOTE – Examples of orchestration of inventory management systems are representing and triggering addition, deletion and changes in the inventory. Benefits of such orchestration mechanisms may include proactive and rapid adaptation of controllers and predictive management of supply chains.

7.38.2 Use case specific figures

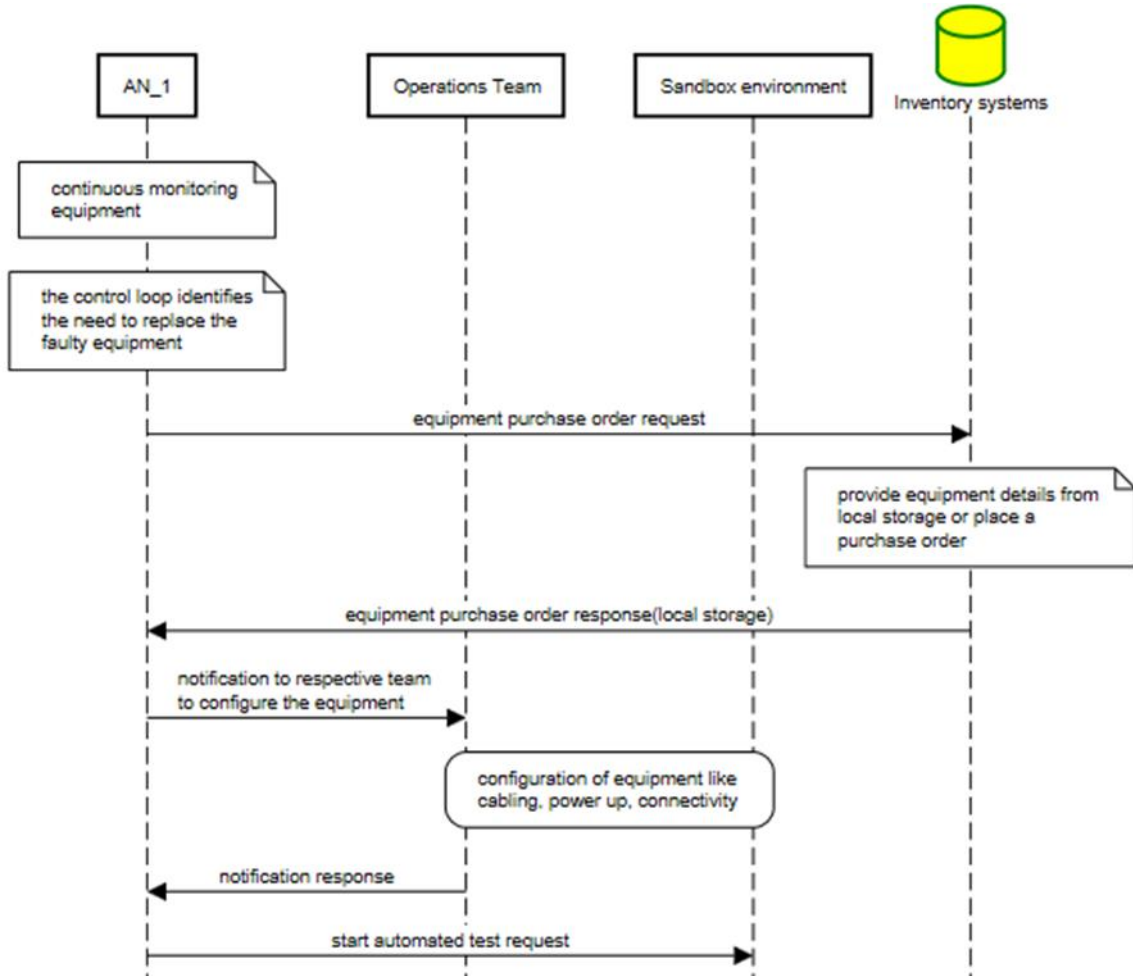


Figure 31 – AN enabled end-to-end supply chain part-1

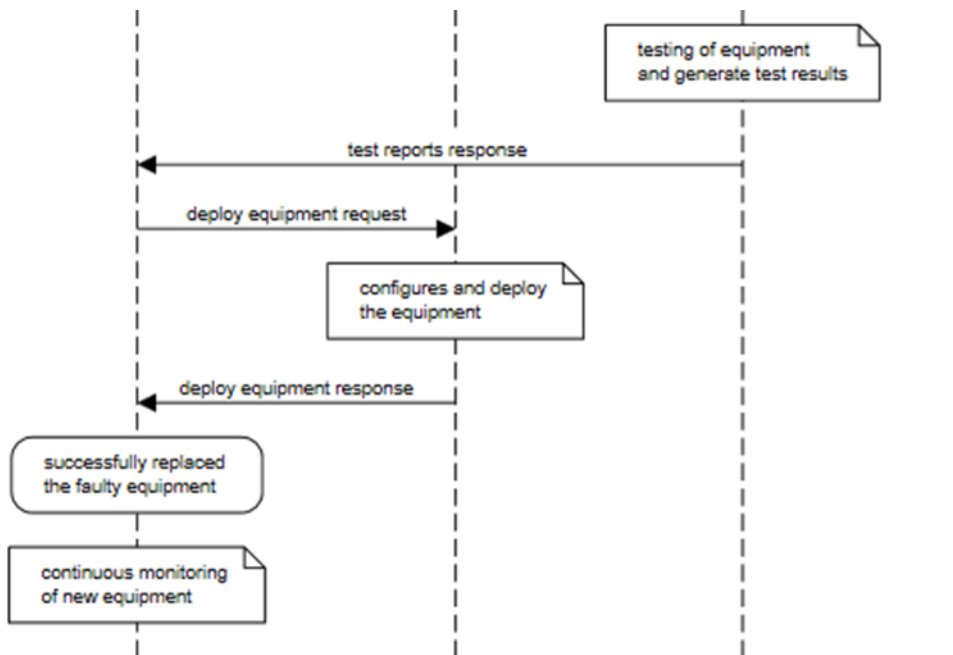


Figure 32 – AN enabled end-to-end supply chain part-2

7.39 Towards openness in AN

<i>Use case id</i>	FG-AN-usecase-039
<i>Use case name</i>	Towards Openness in AN
<i>Base contribution</i>	
<i>Creation date</i>	28/09/2021
<i>Use case context</i>	Inclusion of other control loops in AN
<i>Use case description</i>	<p>We envisage that future networks are heterogeneous AN and control loops are an essential part of AN. There could be various control loops deployed at various endpoints in a heterogeneous AN. The various endpoints could be the edge, cloud, devices like switches, routers, user terminals, customer premises equipment, in general, a control loop possibly in every network equipment. We see that the different vendors would develop various control loops for delivering specific functionality of the AN, focusing on improving cost, efficiency, performance, scalability factors. Also, no single vendor would develop the entire AN. It will be a plug and play design. In such a scenario, there is a need for AN to adopt an open design for inclusiveness. The inclusive design helps the AN to plug any control loop from any vendor based on its requirement. AN act independently without involving any central authority to include the third-party control loops. To enable such inclusive design, the AN shall expose an interface to integrate control loops. The interface requires meta-data about control loops. AN independently manages the lifecycle of the control loops. AN requires the capability to test and configure the control loops. The vendors would use such an interface to develop the application (control loop), knowing that it will help integrate their application. Such a design helps the ecosystem to provide innovative solutions. There could be a control loop application store, where vendors can publish their control loops, and any AN can use it like an android/iOS application or Linux packages. A vendor does not need to publish the control loop in the application store. AN can fetch the control loop from the required location defined in the system.</p> <p>Consider the following example:</p> <ol style="list-style-type: none"> 1. AN identifies the need to deploy a control loop for a particular service or update the existing control (maybe evolutionary). 2. Based on the requirement, AN search the control loop either in the application store or at a pre-configured location. 3. AN download the control loop, test it in a sandbox environment and configures it. The configuration provides control loops providing access to resources. The resources could be data, storage, network, placement, security. 4. Once configured, AN deploys the control loop and continuously monitors the performance of the control loop. 5. AN also configures the SLA and billing settings for the vendor.
<i>Open issues</i>	1. Required interfaces for control loops, billing and SLA
<i>Use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Reference</i>	

7.39.1 Use case requirements

Critical requirements

- AN-UC039-REQ-001: It is critical that autonomous networks support identifying the need for a control loop, enable the selection, evaluation, integration and monitoring of control loops in different underlays.

NOTE – AN may use metadata based selection of control loops from open repositories. Evaluation of control loops may be done using Sandboxes. The control loops are local(internal) or third-party developed.

7.39.2 Use case specific figures

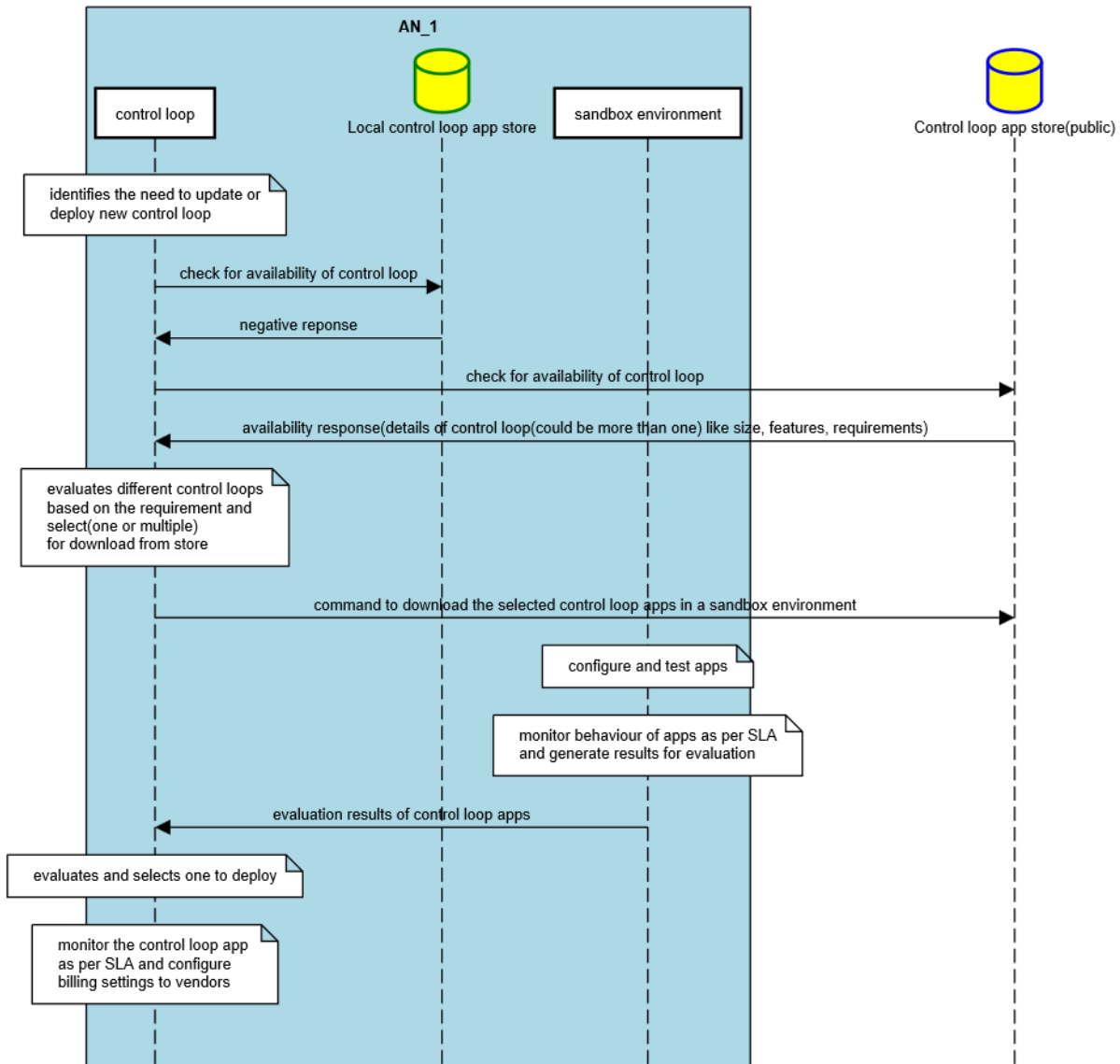


Figure 33 – Towards Openness in AN

7.40 Awareness in AN

<i>Use case id</i>	FG-AN-usecase-040
<i>Use case name</i>	Awareness in AN
<i>Base contribution</i>	
<i>Creation date</i>	28/09/2021
<i>Use case context</i>	Discover and leverage nearby control loops and AN
<i>Use case description</i>	Today we have heterogeneous networks (HetNet) with networks like RAN, Core, transport and convergence of wireless and wireline of different technologies. The evolution of HetNet technologies would comprise control loops in the journey towards AN. Multiple control loops from different vendors and operators surround AN at any given location or endpoint in such a HetNet scenario. AN need to be aware of the surrounding control loops or other AN in general. It could be for various reasons like collaboration, coordination, security, training machine learning model, split learning, offload. AN must have an interface to communicate and interact with other control loops in different AN and control loops within the AN. The interface shall enable other control loops to be autonomous, discover each

	<p>other, communicate directly with other control loops without involving any centralized entity. The control loops within the AN may have a centralized entity, but communication with the external control loop is without a centralized entity. The AN is self-sufficient to act independently. The interface could use the concept presented in FGAN-I-086.</p> <p>Consider the following example of usage of interface:</p> <ol style="list-style-type: none"> 1. The discovery of nearby control loops and AN is available at edge nodes 2. An edge node AN_1 entity running some application and because of heavy burst load in the requests, it runs short of compute and storage to maintain the desired quality of service 3. The AN_1 node identifies the problem based on the feedback, monitoring, any logic/algorithm. It seeks out help in the form of collaboration to offload the requests using the interface 4. Another AN edge node AN_2 responds with its capability and SLA parameters. The AN_1 evaluates and decide to offload the requests to AN_2. 5. AN_1 and AN_2 uses the interface to configure the routing and policy to divert the traffic to AN_2 and initiate the offload process. 6. AN_1 uses the interface to release all the resources and bill the usage as per the SLA once the load reduces
<i>Open issues</i>	<ol style="list-style-type: none"> 1. Required interfaces for control loops, billing and SLA <p>Question – role of taxonomies, ontologies? – if this provides the mechanism to capture the info about the peers, e.g., catalog tagged with the metadata. A flat logical arrangement is possible, which can be operationalized in many ways.</p> <p>Q2: Is peering an absolute must? would a hierarchical structure achieve the same function? Scope of peer "interaction" has to be studied wrt hierarchical, perhaps in adjacent geographic/domains.</p> <p>Q3. Scope of peer interaction has to be granular, flexible, based on dei.</p>
<i>Use case category</i>	Cat 2: Describes a scenario related to application of autonomous behaviour in the network.
<i>Reference</i>	

7.40.1 Use case requirements

Critical requirements

- AN-UC040-REQ-001: It is critical that autonomous networks support peer interaction between different controller instances.

NOTE – Examples of peer interaction are capability discovery and exchange, and resource pooling. Peer interaction may be achieved without involving a centralized entity.

7.40.2 Use case specific figures

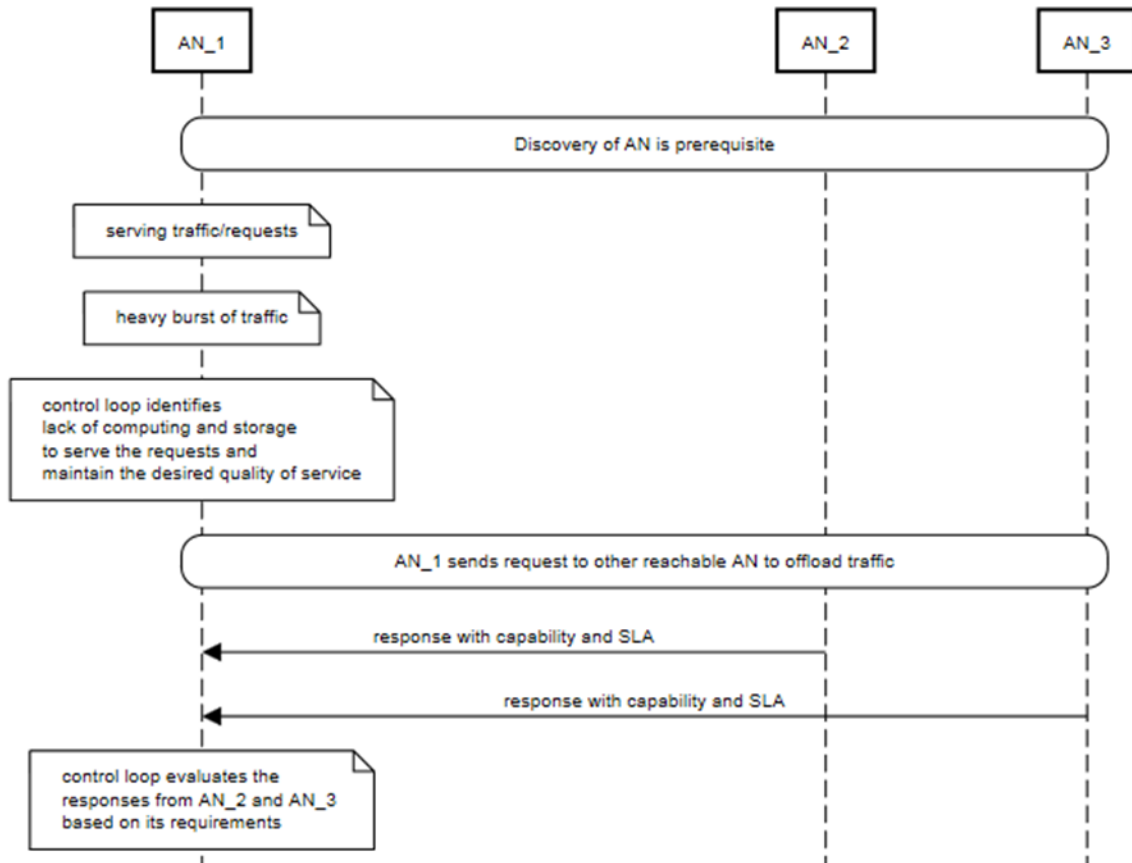


Figure 34 – Actor interaction for Awareness in AN – Part 1

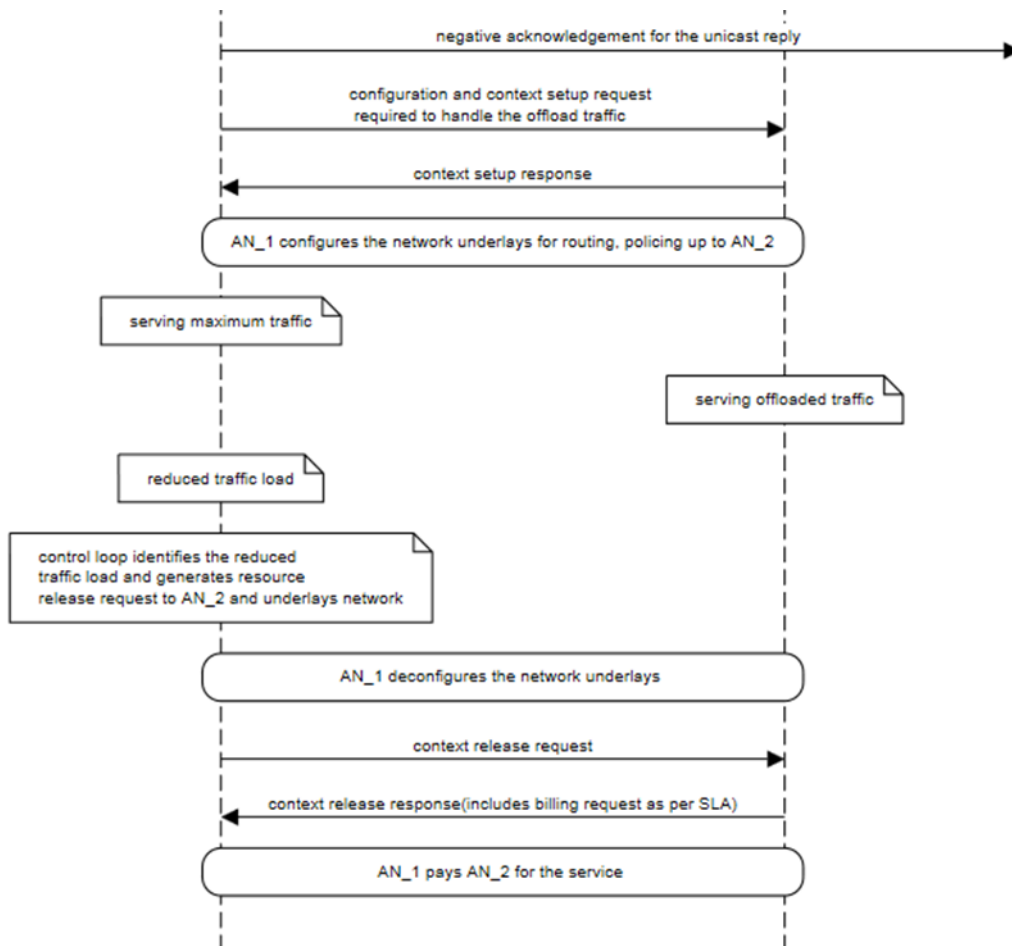


Figure 35 – Actor interaction for Awareness in AN – Part 2

Bibliography

- [b-ITU-T X.1121] Recommendation ITU-T X.1121 (2004), *Framework of security technologies for mobile end-to-end data communications*.
- [b-ITU-T Y.3104] Recommendation ITU-T Y.3104 (2018), *Architecture of the IMT-2020 network*.
- [b-ITU-T Y.3515] Recommendation ITU-T Y.3515 (2017), *Cloud computing – Functional architecture of Network as a Service*.
- [b-ITU-T Y.3525] Recommendation ITU-T Y.3525 (2020), *Cloud computing – Requirements for cloud service development and operation management*.
- [b-Ahmad] Ahmad, I., Narmeen, R., Nguyen, L. D. and Ha, D.-B. (2019), *Quality-of-Service Aware Game Theory-Based Uplink Power Control for 5G Heterogeneous Networks*. Springer Journal of Mobile Networks and Applications, Vol. 24, No. 2, pp. 556–563.
- [b-AI4Good-1] AI for Good Webinar: *Leveraging AI & machine learning to optimize today's 5G radio access network systems and to build the foundation of tomorrow's 6G wireless systems*, 27 November 2020
<https://aiforgood.itu.int/events/leveraging-ai-machine-learning-to-optimize-todays-5g-radio-access-network-systems-and-to-build-the-foundation-of-tomorrows-6g-wireless-systems/>
- [b-AI4Good-2] AI for Good Webinar: *The road towards an AI-native air interface for 6G*, 18 November 2020.
<https://aiforgood.itu.int/events/the-road-towards-an-ai-native-air-interface-for-6g/>
- [b-AI4Good-3] AI for Good Webinar: *Machine learning for joint sensing and communication in future millimeter wave IEEE 802.11 WLANs*, 25 May 2021.
<https://aiforgood.itu.int/events/machine-learning-for-joint-sensing-and-communication-in-future-millimeter-wave-ieee-802-11-wlans/>
- [b-AI4Good-4] AI for Good Webinar: *Service migration algorithms in distributed edge computing systems*, 9 July 2021
<https://aiforgood.itu.int/event/service-migration-algorithms-in-distributed-edge-computing-systems/>
- [b-Almasan] Almasan, P., Suárez-Varela, J., Badia-Sampera, A., Rusek, K., Barlet-Ros, P., & Cabellos-Aparicio, A. (2019). *Deep Reinforcement Learning meets Graph Neural Networks: An optimal routing use case*. arXiv: Networking and Internet Architecture.
- [b-AI-Turjman] Al-Turjman, F., Ever, E. and Zahmatkesh, H. (2019), *Small Cells in the Forthcoming 5G/IoT: Traffic Modelling and Deployment Overview*. IEEE Communications Surveys & Tutorials, Vol. 21, No. 1, pp. 28-65.
- [b-AN2020] ITU Webinar on *Towards a truly Autonomous Network*, 3 November 2020.
<https://www.itu.int/en/ITU-T/webinars/20201103/Pages/default.aspx>

- [b-AVA-1] ITU Technical Report (2013), *Part 2: Vocabulary for ITU-T Focus Group on Audiovisual Media Accessibility (FG AVA)*.
- [b-AVA-2] ITU Technical Report (2013), *Part 3: Using audiovisual media – A taxonomy of participation*.
- [b-Beyerer] Beyerer, J., Deserno, T., Straube, S., Tchouchenkov, I. and Wedler, A. (2021), *Kompetent im Einsatz: Variable Autonomie Lernender Systeme in lebensfeindlichen Umgebungen*. Whitepaper aus der Plattform Lernende Systeme, München.
- [b-Biswas] Biswas, P., Halder, A. Maheshwary, K. and Arjun, S. (2017), *Inclusive Personalization of User Interfaces*. In proceedings of International Conference on Research into Design ICoRD 2017: Research into Design for Communities, Vol. 1, pp 295-306.
- [b-Blott] Blott, M., Preußner, T. B., Fraser, N. J., Gambardella, G., O'brien, K., Umuroglu, Y., Leeser, M. and Vissers, K. (2018), *FINN-R: An end-to-end deep-learning framework for fast exploration of quantized neural networks*. ACM Transactions on Reconfigurable Technology and Systems (TRETS) Vol. 11, No. 3, pp. 1-23.
- [b-Bouraquia] Bouraquia, K., Sabir, E., Sadik, M. and Ladid, L. (2020), *Quality of experience for streaming services: Measurements, challenges and insights*. IEEE Access, Vol. 8, pp. 13341-13361.
- [b-Ciavaglia-1] Ciavaglia, L. (2021), Next-Generation Closed-Loop Automation. IEEE Tutorial, May 17.
<https://www.slideshare.net/LaurentCiavaglia/nextgeneration-closedloop-automation-an-inside-view>
- [b-Ciavaglia-2] Ciavaglia, L., Ghamri-Doudane, S., Smirnov, M., Demestichas, P., Stavroulaki, V.-A., Bantouna, A. and Sayrac, B. (2012), *Unifying Management of Future Networks With Trust*. Bell Labs Technical Journal, Vol. 17, No. 3, pp. 193-212.
- [b-Clark] Clark, D. D., Partridge, C., Ramming, J. C. and Wroclawski, J. T. (2003), *A knowledge plane for the internet*. In Proceedings of the 2003 Conference on Applications, technologies, architectures, and protocols for computer communications, SIGCOMM '03, pp. 3-10.
- [b-DISH-AWS] Businesswire (2021), *DISH and AWS Form Strategic Collaboration to Reinvent 5G Connectivity and Innovation*.
<https://www.businesswire.com/news/home/20210421005315/en/DISH-and-AWS-Form-Strategic-Collaboration-to-Reinvent-5G-Connectivity-and-Innovation>
- [b-ETSI GR ZSM 009-3] Draft Group Report ETSI GR ZSM 009-3, *Zero-Touch Network and Service Management (ZSM) Closed-loop automation: Advanced topics*.
- [b-ETSI GS ZSM 001] Group Specification ETSI GS ZSM 001 (2019), *Zero-touch network and Service Management (ZSM); Requirements based on documented scenarios*.

- [b-ETSI GS ZSM 002] Group Specification ETSI GS ZSM 002 (2019), *Zero-touch network and Service Management (ZSM); Reference Architecture*.
- [b-ETSI GS ZSM 013] Draft Group Specification ETSI GS ZSM 013, *Zero-touch network and Service Management (ZSM); Automation of CI/CD for ZSM services and managed services*.
- [b-FLAML] FLAML Demo (2021), *Fast and Lightweight AutoML – a brief overview*.
<https://youtu.be/YRbLmDirjsc>
- [b-Flightplan] Flightplan. <https://flightplan.cis.upenn.edu/>
- [b-FRINX-1] FRINX Machine Introduction.
<https://docs.frinx.io/frinx-machine/getting-started/getting-started-with-frinx-machine.html>
- [b-FRINX-2] FRINX Github repository, <https://github.com/FRINXio>.
- [b-Hesse] Hesse, T., Peylo, C., Balhmann, C., Elbe, A. Camargo, I. N. and Slusallek, P. (2021), *Potenziale für industrieübergreifendes Flottenlernen KI-Mobilitätsdatenplattform zur Risikominimierung des automatisierten Fahrens*. Whitepaper aus der Plattform Lernende Systeme, München.
- [b-ISO 9241-129] ISO 9241-129 (2010), *Ergonomics of human-system interaction – Part 129: Guidance on software individualization*.
- [b-ISO/IEC 23167] ISO/IEC TS 23167 (2020), *Information technology – Cloud computing – Common technologies and techniques*.
- [b-ISO/IEC 24756] ISO/IEC 24756 (2009), *Information technology – Framework for specifying a common access profile (CAP) of needs and capabilities of users, systems, and their environments*.
- [b-J.acc-us-prof] Draft Recommendation J.acc-us-prof, *Common user profile format for audiovisual content*.
- [b-Jagannath] Jagannath, J., Polosky, N., Jagannath, A., Restuccia, F., and Melodia, T. (2019), *Machine Learning for Wireless Communications in the Internet of Things: A Comprehensive Survey*. Ad Hoc Networks Vol. 93, p. 101913.
- [b-Jahromi] Jahromi, H. and Delaney, D. (2018), *An Application Awareness Framework Based on SDN and Machine Learning: Defining the Roadmap and Challenges*. In 13th APCA International Conference on Control and Soft Computing (CONTROLO), pp. 411-416.
- [b-Jimenez-Ruiz] Jimenez-Ruiz, E. (2021), *Semantic Web Technologies and Knowledge Graphs*. Teaching notes, University of London.
<https://github.com/turing-knowledge-graphs/teaching/tree/main/city>
- [b-KUKA] KUKA – Robots & Automation (2021), *Robot AI makes robots safer for robot-human collaboration, even without programming*.
https://www.youtube.com/watch?v=e0eKrM0_oyQ

- [b-Liu] Liu, L., Hu, H., Luo, Y. and Wen, Y. (2020), *When wireless video streaming meets AI: A deep learning approach*. IEEE Wireless Communications, Vol. 27, No. 2, pp. 127–133.
- [b-Mao] Mao, Q., Hu, F. and Hao Q. (2018), *Deep Learning for Intelligent Wireless Networks: A Comprehensive Survey*. In IEEE Communications Surveys & Tutorials, Vol. 20, No. 4, pp. 2595-2621.
- [b-Microsoft] Microsoft (2021), FLAML – Fast and Lightweight AutoML, <https://github.com/microsoft/FLAML>
- [b-Myklebust] Myklebust, E. B., Jiménez-Ruiz, E., Chen, J., Wolf, R. and Tollefsen, K. E. (2021), *Prediction of Adverse Biological Effects of Chemicals Using Knowledge Graph Embeddings*. Semantic Web Journal, 0 (0) 1, IOS Press.
- [b-Nahum] Nahum, C. V., Pinto, L., Tavares, V. B., Batista, P., Lins, S., Linder, N. and Klautau, A. (2020), *Testbed for 5G Connected Artificial Intelligence on Virtualized Networks*. IEEE Access, Vol. 8, pp. 223202-223213.
- [b-ONF] The Open Networking Foundation (ONF) (2021), *The Open Networking Foundation, AirHop, Facebook and Telecom Infra Project Demonstrate First O-RAN Aligned, xApp-Powered Open RAN Solution*. https://opennetworking.org/news-and-events/press-releases/the-open-networking-foundation-airhop-facebook-and-telecom-infra-project-demonstrate-first-o-ran-aligned-xapp-powered-open-ran-solution/?utm_source=ONF+Master+List&utm_campaign=b8631125bb-Edge+Spot
- [b-O'Shea-1] O'Shea, T. J. and Hoydis, J. (2017), *An Introduction to Deep Learning for the Physical Layer*. IEEE Transactions on Cognitive Communications and Networking, Vol. 3, No. 4, pp. 563–575.
- [b-O'Shea-2] O'Shea, T. J., Roy, T. and Clancy, T. C. (2018), *Over-the-Air Deep Learning Based Radio Signal Classification*. IEEE Journal of Selected Topics in Signal Processing, Vol. 12, No. 1, pp. 168-179.
- [b-Pierucci] Pierucci, L. (2015), *The quality of experience perspective toward 5g technology*. IEEE Wireless Communications, Vol. 22, No. 4, pp. 10-16.
- [b-Q.IMT2020-PIAS] Draft Recommendation ITU-T Q.IMT2020-PIAS, *Protocol for providing intelligent analysis services in IMT-2020 network*.
- [b-Rusek] Rusek, K., Suárez-Varela, J., Mestres, A., Barlet-Ros, P. and Cabellos-Aparicio, A., (2019), *Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN*. In Proceedings of the 2019 ACM Symposium on SDN Research (SOSR '19), pp. 140-151.
- [b-Sankar-1] Sankar, V. U. and Sharma, V. (2016), *QoS Provisioning for Multiple Femtocells via Game Theory*. Elsevier Journal of Computer Networks, Vol 102, pp.70-82.

- [b-Sankar-2] Sankar, V. U. and Sharma, V. (2014), *Pricing and Power allocation in Femtocells using Stackelberg Game Theory*. In: International Conference on Signal Processing and Communications (SPCOM), Jul 22-25, Bangalore, India.
- [b-SCALE-Sim] SCALE-Sim tutorial – ASPLOS 2021.
<https://scalesim-project.github.io/tutorials-2021-asplos.html>
- [b-Sharma] Sharma, R. (2020), *Rakuten Mobile to Use Drones to Carry Out Mobile Base Station Inspections*.
<https://www.thefastmode.com/technology-solutions/16814-rakuten-mobile-to-use-drones-to-carry-out-mobile-base-station-inspections>
- [b-TIP 5G] TIP 5G Open Automation: Vision of Reality, July 2021.
<https://hopin.com/events/tip-5g-open-automation-vision-of-reality>
- [b-Turing] The Alan Turing Institute (2021), *Knowledge graphs*.
<https://www.turing.ac.uk/research/interest-groups/knowledge-graphs>
- [b-Umuroglu] Umuroglu, Y., Akhauri, Y., Fraser, N. J. and Blott, M. (2020), *LogicNets: Co-Designed Neural Networks and Circuits for Extreme-Throughput Applications*. In 30th International Conference on Field-Programmable Logic and Applications (FPL), pp. 291-297.
- [b-Wang-1] Wang, C., Wu, Q., Weimer, M. and Zhu, E. (2021), *FLAML: A Fast and Lightweight AutoML Library*. In Fourth Conference on Machine Learning and Systems (MLSys 2021).
- [b-Wang-2] Wang, C., Wu, Q., Huang, S. and Saied, A. (2021), *Economical Hyperparameter Optimization With Blended Search Strategy*. In the Ninth International Conference on Learning Representations (ICLR 2021).
- [b-Wu-1] Wu, Q., Wang, C. and Huang, S. (2021), *Frugal Optimization for Cost-related Hyperparameters*. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 12: AAAI-21.
- [b-Wu-2] Wu, Q., Wang, C., Langford, J., Mineiro, P. and Rossi, M. (2021), *ChaCha for Online AutoML*. In Proceedings of the 38th International Conference on Machine Learning, PMLR 139, pp. 11263-11273.
- [b-Xilinx] Xilinx (2021), *Accelerate Your AI-Enabled Edge Solution with Adaptive Computing, Introducing Adaptive System-on-Modules (SOMs)*.
- [b-Y.ML-IMT2020-MLFO] Draft Recommendation ITU-T Y.ML-IMT2020-MLFO, *Requirements and architecture for machine learning function orchestrator*.
- [b-Y.ML-IMT2020-SANDBOX] Draft Recommendation ITU-T Y.ML-IMT2020-SANDBOX, *Machine learning sandbox for future networks including IMT-2020: requirements and architecture framework*.