

Oracle® Communications

Networks Data Analytics Function

Benchmarking Guide



Release 24.2.2
F96694-02
November 2024

ORACLE®

Copyright © 2022, 2024, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

1	Introduction	
	1.1 Purpose and Scope	1
	1.2 References	1
2	Deployment	
3	Resource Requirement	
4	OCNWDAF Benchmarking Testing	

My Oracle Support

My Oracle Support (<https://support.oracle.com>) is your initial point of contact for all product support and training needs. A representative at Customer Access Support can assist you with My Oracle Support registration.

Call the Customer Access Support main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at <http://www.oracle.com/us/support/contact/index.html>. When calling, make the selections in the sequence shown below on the Support telephone menu:

- For Technical issues such as creating a new Service Request (SR), select **1**.
- For Non-technical issues such as registration or assistance with My Oracle Support, select **2**.
- For Hardware, Networking, and Solaris Operating System Support, select **3**.

You are connected to a live agent who can assist you with My Oracle Support registration and opening a support ticket.

My Oracle Support is available 24 hours a day, 7 days a week, 365 days a year.

Acronyms

The following table provides information about the acronyms and the terminology used in the document.

Table Acronyms

Acronym	Description
3GPP	3rd Generation Partnership Project
5GC	5G Core Network
5GS	5G System
AF	Application Function
API	Application Programming Interface
AMF	Access and Mobility Management Function
CNC	Cloud Native Core
CNE	Oracle Communications Cloud Native Core, Cloud Native Environment (CNE)
FQDN	Fully Qualified Domain Name
GUI	Graphical User Interface
HTTPS	Hypertext Transfer Protocol Secure
KPI	Key Performance Indicator
HA	High Availability
IMSI	International Mobile Subscriber Identity
K8s	Kubernetes
ME	Monitoring Events
Network Slice	A logical network that provides specific network capabilities and network characteristics.
NEF	Oracle Communications Cloud Native Core, Network Exposure Function (NEF)
NF	Network Function
NRF	Oracle Communications Cloud Native Core, Network Repository Function (NRF)
NSI	Network Slice Instance. A set of Network Function instances and the required resources (such as compute, storage and networking resources) which form a deployed Network Slice.
NSSF	Oracle Communications Cloud Native Core, Network Slice Selection Function (NSSF)
NWDAF	Network Data Analytics Function
OAM	Operations, Administration, and Maintenance
PLMN	Public Land Mobile Network
REST	Representational State Transfer
SBA	Service Based Architecture
SBI	Service Based Interface
SMF	Session Management Function
SNMP	Simple Network Management Protocol
SUPI	Subscription Permanent Identifier
UDM	Unified Data Management

Table (Cont.) Acronyms

Acronym	Description
UE	User Equipment
URI	Uniform Resource Identifier

What's New in This Guide

This section introduces the documentation updates for Release 24.2.x in *Oracle Communications Networks Data Analytics Function Benchmarking Guide*.

Release 24.2.2 - F96694-02, November 2024

There are no updates in this document.

Release 24.2.0 - F96694-01, July 2024

Updated the following sections:

- Updated the [Purpose and Scope](#) section with OCI information.
- Updated the core microservice requirements in the [Resource Requirement](#) section.

1

Introduction

Oracle Communications Networks Data Analytics Function (OCNWDAF) is a Network Function (NF) that assists in collecting and analyzing data in a 5G network.

The 5G technology requires prescriptive analytics to drive closed-loop automation and self-healing networks. In a 5G network, the consumers of data are 5G NFs, Application Functions (AFs), and Operations, Administration, and Maintenance (OAM) and the data producers are 5G NFs. An OCNWDAF consumer can avail analytics information for different analytic events. Alternatively, the consumers can subscribe or unsubscribe for specific analytics information as a one-time event or periodically get notified when a specifically defined event (for example, a threshold is breached) is detected.

1.1 Purpose and Scope

This document is designed to help operators measure the capacity and performance of OCNWDAF, OCNWDAF microservices, and deployment environment setup software such as Cloud Native Environment (CNE), Oracle Cloud Infrastructure (OCI), and cnDBTier.

This document provides OCNWDAF performance and capacity data.

It is recommended that OCNWDAF is run through a benchmark on the target cloud native infrastructure to determine the capacity and performance in the target infrastructure. This information can be used to adjust the initial deployment resources and to help predict resource requirements when OCNWDAF is scaled up.

1.2 References

For more information about OCNWDAF, see the following documents:

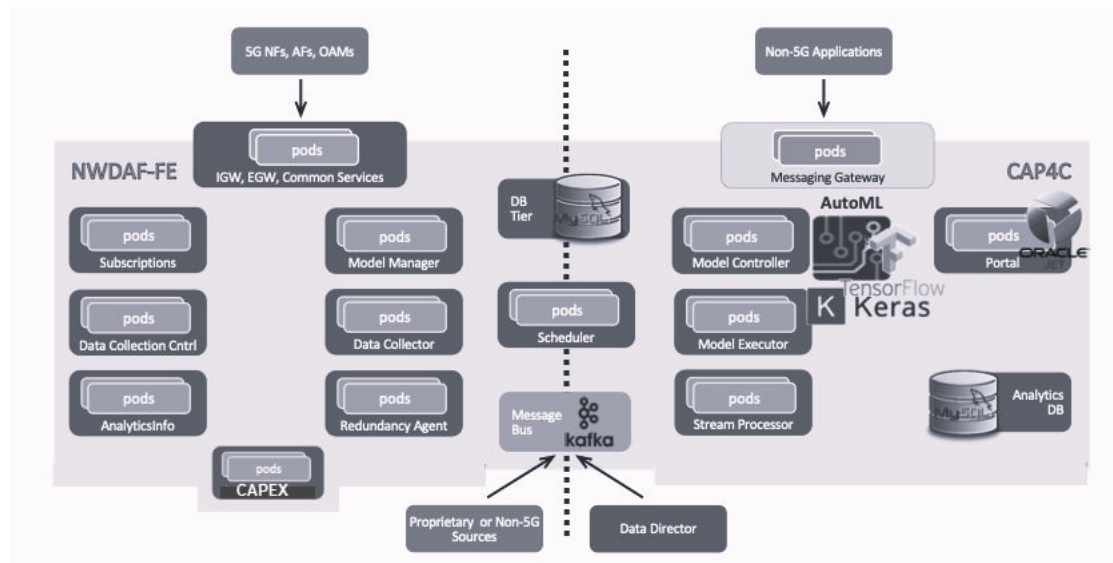
- *Oracle Communications Networks Data Analytics Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Networks Data Analytics Function User Guide*
- *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, cnDBTier Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Network Analytics Automated Testing Suite Guide*

2

Deployment

OCNWDAF can be deployed in the CNE environment. For more information about OCNWDAF installation, see *Oracle Communications Networks Data Analytics Function Installation and Fault Recovery Guide*. The following diagram depicts the OCNWDAF deployment in the 5G architecture:

Figure 2-1 OCNWDAF in a 5G Network



OCNWDAF uses the following common services of CNE:

- Kubernetes
- HELM
- cnDBTier
- CNC Console
- Prometheus
- Metallb (Load balancer)

3

Resource Requirement

This chapter provides information about the resource requirements to install and run Oracle Communications Networks Data Analytics Function (OCNWDAF).

Kubernetes Resources

The following table displays the Kubernetes resource requirements:

Table 3-1 Kubernetes Resources

Instance	No. of instances	CPU/Instance	Total CPU
K8s Master	3	4	12
K8s Worker	12	16	192
LB	2	2	4
Total			208

OCNWDAF Deployment

The following table displays the OCNWDAF deployment details:

Table 3-2 OCNWDAF Deployment

Deployment Type	CPUs	Memory (in GB)	Storage (in GB)
Default	105	969	5000

Microservice Port Mapping

The following table displays the microservices port mapping information:

Table 3-3 Port Mapping

Service	Port Type	IP Type	Network Type	Service Port	Container Port
ocn-nwdaf-analytics	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
nwdaf-egress-gateway	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
nwdaf-ingress-gateway	External	NodePort	External/ K8s	80/TCP	8081/TCP
ocn-nwdaf-data-collection-service	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
ocn-nwdaf-datacollection-controller	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
ocn-nwdaf-mtlf	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP

Table 3-3 (Cont.) Port Mapping

Service	Port Type	IP Type	Network Type	Service Port	Container Port
ocn-nwdaf-subscription-service	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
ocn-nwdaf-analytics-info	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
ocn-nwdaf-georedagent	Internal	ClusterIP	Internal / K8s	9181/TCP	9181/TCP
cap4c-kafka-ingestor	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
cap4c-model-controller	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
cap4c-model-executor	Internal	ClusterIP	Internal / K8s	9092/TCP	9092/TCP
cap4c-stream-transformer	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
cap4c-stream-analytics	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
cap4c-api-gateway	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
nwdaf-cap4c-reporting-service	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
nwdaf-cap4c-scheduler-service	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
nwdaf-cap4c-spring-cloud-config-server	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
nwdaf-portal	External	NodePort	External / K8s	80/TCP	
nwdaf-portal-service	Internal	ClusterIP	Internal / K8s	8080/TCP	8080/TCP
cap4c-configuration-manager-service	Internal	ClusterIP	Internal / K8s	9000/TCP	9000/TCP
cap4c-capex-optimization-service	Internal	ClusterIP	Internal / K8s	9000/TCP	9000/TCP

OCNWDAF Core Microservices Requirements

The following table displays the core microservices resource requirements:

Table 3-4 Core Microservices Resource Requirements

Microservice Name	Instances	POD Replica		CPU/POD		Memory/POD (in GB)		Ephemeral Storage	
		Min	Max	Min	Max	Min	Max	Min (Mi)	Max (GB)

Table 3-4 (Cont.) Core Microservices Resource Requirements

Microservice Name	Instances	POD Replica		CPU/POD		Memory/POD (in GB)		Ephemeral Storage	
ocn-nwdaf-analytics-info-service	1	1	2	1	2	1	2	200	200
nwdaf-ingress-gateway	2	2	5	1	2	2	4	200	200
nwdaf-egress-gateway	2	2	5	1	2	2	4	200	200
nwdaf-cap4c-spring-cloud-config-server	1	1	1	1	2	1	2	200	200
ocn-nwdaf-data-collection-service	1	1	3	4	4	4	4	200	200
ocn-nwdaf-data-collection-controller	1	1	2	1	2	1	2	200	200
ocn-nwdaf-subscription-service	1	1	2	1	2	1	2	200	200
ocn-nwdaf-mtlf-service	1	1	2	1	2	1	2	200	200
cap4c-configuration-manager-service	1	1	2	1	2	1	2	200	200
cap4c-model-controller	1	1	3	1	2	1	2	200	200
cap4c-model-executor	1	1	3	1	2	1	2	200	200
cap4c-stream-analytics	1	1	3	1	2	1	2	200	200
nwdaf-portal	1	1	2	0.5	1	1	2	200	200
nwdaf-portal-service	1	1	2	1	2	1	2	200	200
cap4c-scheduler-service	1	1	2	1	2	1	2	200	200
cap4c-stream-transformer	1	1	2	1	2	1	2	200	200
cap4c-api-gateway	1	1	2	1	2	1	2	200	200
cap4c-kafka-ingestor	1	1	3	1	2	1	2	200	200
ocn-nwdaf-cap4c-reporting-service	1	1	2	1	2	1	2	200	200
ocn-nwdaf-geo-redundancy-agent	1	1	2	1	1	1	1	200	200
nwdaf-cap4c data-replicator	1	1	1	1	1	2	4	200	200

Table 3-4 (Cont.) Core Microservices Resource Requirements

Microservice Name	Instances	POD Replica		CPU/POD		Memory/POD (in GB)		Ephemeral Storage	
cap4c-capex-optimization-service	1	1	2	1	2	1	2	200	200
ocn-nwdaf-analytics-decision-engine-service	1	1	2	1	2	1	2	200	200
Total	25	25	55	12.5	45	30	55	4600	4600

NRF Client Resources

The following table displays the NRF Client resource requirements:

Table 3-5 NRF Client Resource Requirements

Microservice Name	POD Replica		CPU/POD		Memory/POD (in GB)		Ephemeral Storage	
	Min	Max	Min	Max	Min	Max	Min (Mi)	Max (G)
nrfclient-appinfo	1	2	1	2	1	2	78.1	1
nrfclient-ocnf-nrf-client-discovery (2 instances)	1	2	1	2	1	2	78.1	1
nrfclient-ocnf-nrf-client-nfmanagement	1	2	1	2	1	1	78.1	1
nrfclient-ocpm-config	1	2	1	2	1	4	78.1	1
Total	4	8	4	8	4	9		

ATS and Stubs Requirements

The following table displays the ATS and Stubs resource requirements:

Table 3-6 ATS and Stubs Requirements

Microservice	vCPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	Replicas (regular deployment)	Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
ocn-ats-nwdaf-service	4	8	0	1	1	4	8	0
ocn-ats-nwdaf-notify-service	2	2	0	1	1	2	2	0

CNC Console Requirements

The following table displays the CNC Console resource requirements:

Table 3-7 CNC Console Requirements

Service Name	Limits		Requests	
	CPU	Memory (in GB)	CPU	Memory (in GB)
M-CNCC IAM	10.5	11.5	5.3	5.8

Redis Requirements

The following table displays the Redis requirements:

Table 3-8 Redis Requirements

Microservice Name	POD Replica		CPU/POD		Memory/POD (in GB)		Storage	
	Min	Max	Min	Max	Min	Max	Min (Mi)	Max (G)
Redis	1	3	2	4	4	8	500	1
Total	1			4		8	500	

cnDBTier Requirements

For information on cnDBTier requirements see, *Oracle Communications Cloud Native Core, cnDBTier Installation, Upgrade, and Fault Recovery Guide*.

Kafka Requirements

The following tables display the Kafka requirements:

Each entry is per broker (with at least three brokers and three zookeepers).

Table 3-9 Broker Configuration

Broker Configuration	Brokers	CPU/POD	RAM (G)	Storage (G)
Basic	4	1	32	500
Default	12	12	64	1000

With a minimum of three nodes in the cluster, the expected data transfer rate is 225 MB/sec.

MySQL Innodb Cluster

The following table displays the MySQL Innodb Cluster requirements:

Table 3-10 MySQL Innodb Cluster Requirements

Microservice Name	Pod Replica		CPU/Pod		Memory/Pod(in G)		Total	
	Min	Max	Min	Max	Min	Max	Min(Mi)	Max(G)

Table 3-10 (Cont.) MySQL Innodb Cluster Requirements

Microservice Name	Pod Replica		CPU/Pod		Memory/Pod(in G)		Total	
nwdaf-mysql-innodb-cluster-router	1	2	1	2	1	2	78.1	1
nwdaf-mysql-innodb-cluster	1	3	1	2	1	3	78.1	1
nwdaf-mysql-operator	1	1	1	2	1	1	78.1	1
	3	6	4	6	3	6		

4

OCNWDAF Benchmarking Testing

This chapter describes various testing scenarios and the results obtained by running performance tests on Oracle Communications Networks Data Analytics Function.

A series of scripts is created to simulate the entire flow of CAP4C model execution and to extract the performance metrics of the models created. The scripts perform the following tasks:

- Insert synthetic data into the database.
- Call the Model Controller API to train the models and perform the prediction tasks.
- Retrieve the metrics from Jaeger.
- Delete the synthetic data in the database.

The table below displays the expected performance metrics (per model) on completion of the flow of CAP4C model executions:

Table 4-1 Expected Performance Metrics

Metric Name	Expected Value
UE Mobility models trained	6
UE Mobility model trained max time	275202.415ms
UE Mobility model trained avg time	56976.9326ms
UE Mobility model trained min time	2902.839ms
Slice Load models trained	10
Slice Load model trained max time	11345.23
Slice Load model trained avg time	10212.96
NF Load models trained	5
NF Load model trained max time	9606.666 ms
NF Load model trained avg time	8852.0124 ms
NF Load model trained min time	8272.789 ms
Abnormal behavior models trained	8
Abnormal behavior m model trained max time	8297.173 ms
Abnormal behavior m model trained avg time	7504.53 ms
Abnormal behavior m model trained min time	7094.006 ms
Network Performance model trained	4
Network Performance model trained max time	245966 ms
Network Performance model trained avg time	69722.4 ms
Network Performance model trained min time	8894 ms
QoS GBR models trained	3
QoS GBR models trained max time	3539.15 ms
QoS GBR models trained avg time	1196.60 ms
QoS GBR models trained min time	17.48 ms
QoS Non-GBR models trained	3
QoS Non-GBR models trained max time	3478.50 ms

Table 4-1 (Cont.) Expected Performance Metrics

Metric Name	Expected Value
QoS Non-GBR models trained avg time	1170.91 ms
QoS Non-GBR models trained min time	24.70 ms

Script Execution

The following two scripts are created to simulate the CAP4C model execution and to extract the performance metrics of the models:

- SQL script to insert data and call the Model Controller API.
- Script to extract Jaeger Metrics and delete synthetic data from the database.

A docker image is provided with the scripts, it can be deployed in the Kubernetes cluster where all the OCNWDAF services are running. The docker image is provided in the images folder of OCNWDAF installer.

- To create the values.yaml file, use the following information:

```
global:
  projectName: nwdaf-cap4c-initial-setup-script
  imageName: nwdaf-cap4c/nwdaf-cap4c-initial-setup-script
  imageVersion: 24.1.0.0.0
config:
  env:
    APPLICATION_NAME: nwdaf-cap4c-initial-setup-script
    APPLICATION_HOME: /app
deploy:
  probes:
    readiness:
      enabled: false
    liveness:
      enabled: false
```

- To deploy and run the image, create the values.yml file, run the following command:

```
$ helm install <deploy_name> <helm_chart> -f <path_values_file>/values.yml
-n <namespace_name>
```

For example:

```
$ helm install nwdaf-cap4c-initial-setup-script https://artifactory-
phx.oci.oraclecorp.com/artifactory/ocnwdafe-helm/nwdaf-cap4c-deployment-
template-23.2.0.tgz -f <path_values_file>/values.yml -n <namespace_name>
```

- A container will be running inside the Kubernetes cluster. Identify the container and note down the name of the container for reference in subsequent steps. Run the following command to obtain pod information:

```
$ kubectl get pods -n <namespace_name>
```

For example:

```
NAME                                READY   STATUS    RESTARTS   AGE
nwdaf-cap4c-initial-setup-script-deploy-64b8fbcd9-2vqf9   1/1
Running                                0        55s
```

- Run the following command to access the container:

```
$ kubectl exec -n <namespace_name> <pod_name> -it bash
```

- Once inside the container, navigate to the path /app/performance. The scripts are located in this path, follow the steps below to run the scripts:
 - SQL script to insert data and to call the model controller api:** This script handles the insertion of the synthetic data into the DB. Once all the tables are inserted, a process runs to call the Model Controller API. The models are generated and execution tasks are created to use the models. Run the script:

```
$ python3 performance/integration_script.py -h <host> -p <path> -t
<type> -n <number_tests>
```

Table 4-2 Parameters

Parameter	Description	Default Value
-h	The Model controllers host name or address	localhost
-p	Path of the .csv files that have to create the request to controller	../integration/data/
-t	The files extension to read the data to generate payloads	CSV
-n	Number of tests flows that the script will do. The recommended value of number of tests to be performed per pod.	10

Note

If the parameters are not set, the script uses the default values.

Example:

```
$ python3 integration_script.py -h http://cap4c-model-
controller.ocnwdaf-ns:8080 -p /app/performance/integration/data/ -t CSV
-n 10
```

Sample output:

Figure 4-1 Sample Output

```

Model: ABNORMAL_BEHAVIOUR
Request type: EXECUTION
Models: 16
Min time: 2005.943ms
Max time: 10706.774ms
Avg time: 6298.1315ms
Request type: TRAIN
Models: 8
Min time: 7094.006ms
Max time: 8297.173ms
Avg time: 7504.525625ms
Model: NF_LOAD
Request type: EXECUTION
Models: 10
Min time: 3177.181ms
Max time: 6255.129ms
Avg time: 4704.9093ms
Request type: TRAIN
Models: 5
Min time: 8272.789ms
Max time: 9606.666ms
Avg time: 8852.0124ms

```

- **Script to extract Jaeger Metrics and to subsequently delete synthetic data from Database:** This script extracts the metrics and then deletes the dummy data from the DB. Run the script:

```
$ python3 performance/report_script.py -h <host> -p <port> -m
<options> -u <prefix> -y <types>
```

Table 4-3 Parameters

Parameter	Description	Default Value
-h	Jaeger host name or address	localhost
-p	Jaeger UI port	16686
-m	Available options: ABNORMAL_BEHAVIOUR, NF_LOAD, and so on.	all
-u	Set a URL prefix if needed, such as '/blurr7/jaeger'	NA
-y	Type of execution: TRAIN, EXECUTION, all.	all

Note

If the parameters are not set, the script uses the default values.

For example:

```
$ python report_script.py -t occne-tracer-jaeger-query.occne-infra -p
80 -u /blurr7/jaeger
```

Sample output:

Figure 4-2 Sample Output

```
Model: UE_MOBILITY
Request type: EXECUTION
Models: 0
Min time: 0.0ms
Max time: 0.0ms
Avg time: 0.0ms
Request type: TRAIN
Models: 8
Min time: 5564.09ms
Max time: 11002.851ms
Avg time: 8555.22675ms
```

Figure 4-3 Sample Output

```
Model: ABNORMAL_BEHAVIOUR
Request type: EXECUTION
Models: 16
Min time: 2005.943ms
Max time: 10706.774ms
Avg time: 6298.1315ms
Request type: TRAIN
Models: 8
Min time: 7094.006ms
Max time: 8297.173ms
Avg time: 7504.525625ms
Model: NF_LOAD
Request type: EXECUTION
Models: 10
Min time: 3177.181ms
Max time: 6255.129ms
Avg time: 4704.9093ms
Request type: TRAIN
Models: 5
Min time: 8272.789ms
Max time: 9606.666ms
Avg time: 8852.0124ms
```

Network Performance Analytics

The Network Performance Analytics provides various insights about the performance of a network. This information can be used for devising and deciding network strategies. The Network Performance data collected by the OCNWDAF can be used for network optimization, optimal resource allocation, and service assurance.

Predictive Analytics

The predictive analysis of Network Performance parameters in OCNWDAF facilitates the implementation of preemptive actions for improving the Network Performance by predicting values of attributes that impact the performance of a network.

Machine Learning (ML) Models and Performance Metrics

Predictive analysis of Network Performance parameters is based on time series prediction of Network Performance attributes:

Table 4-4 Network Performance Attributes

Attributes	Input or Output to Model
TimeStamp	Input
sessSuccRatio	Output (predicted)
hoSuccRatio	Output (predicted)
gnbComputingUsage	Output (predicted)
gnbMemoryUsage	Output (predicted)
gnbDiskUsage	Output (predicted)
gnbActiveRatio	Output (predicted)

The Long Short-Term Memory (LSTM) algorithm is a widely used Machine Learning (ML) algorithm for time-series prediction. Four LSTM models with some variations in their tuning are used for predicting the Network Performance parameters in OCNWDAF and integrated with CAP4C. The following table provides information on the base model, training time, and tuning parameters:

Table 4-5 LSTM

Base Model	Training Time (MS)	Tuning Parameters		
		Epochs	Optimizer Used	Batch Size
LSTM	245966	100	Adm	1
LSTM	8894	20	Sgd	32
LSTM	11840	50	AdaMax	32
LSTM	12189.6	50	Adadelta	32

Quality of Service (QoS) Sustainability Analytics

The Quality of Service (QoS) Sustainability Analytics information includes QoS change statistics in a target area for a specified period. Improving QoS in a network comprises reducing packet loss and latency by managing network data and traffic. Quality of Service (QoS) Sustainability Analytics information is used to manage network resources optimally.

QoS Predictions for GBR

Predictive analytics includes information about anticipated QoS changes at a future time in a target area. It leverages information obtained by historic network service usage practices.

Machine Learning Models and Performance Metrics Per QoS Model

The attributes listed in the table below are used for QoS predictions:

Table 4-6 Attributes

Attributes	Input/Output to Model
load_timestamp	Input
snssai	Input
nr_cell_id	Input
5qi	Input
number_of_released_active_qos_flow	Output (QoS prediction)

Machine Learning (ML) Models for QoS Prediction and Performance Metrics

Listed below are the ML models used for QoS predictions and performance metrics:

- Random Forest Regressor
- K Nearest Neighbour (KNN)
- Decision Tree

QoS Prediction Models

Table 4-7 QoS Prediction Models

Quality of Service (QoS) Prediction	
Model/Algorithm	Training Time in Millisec.
Random Forest Regrec ssor	3539.15
K Nearest Neighbour (KNN)	17.48
Decision Tree	33.18
Model Training Time and Total Models	
Total Models	3
Minimum Time (ms)	17.48
Maximum Time (ms)	3539.15
Average Time (ms)	1196.60

QoS Predictions for Non-GBR

The Down Link Throughput is an important and effective performance indicator for determining the user experience. Hence, the prediction of Down Link Throughput plays a vital role in network dimensioning.

ML Models and Performance Metrics Per DL Throughput Model

The attributes listed in the table below are used for DL Throughput prediction:

Table 4-8 Attributes

Attributes	Input/Ouput to Model
load_timestamp	Input
snssai	Input
nr_cell_id	Input
5qi	Input
downlink_throughput	Output (DL Throughput prediction)

Machine Learning (ML) Models for QoS Predictions for Non-GBR Prediction and Performance Metrics

Listed below are the ML models used for QoS predictions and performance metrics:

- Random Forest Regressor
- K Nearest Neighbour (KNN)
- Decision Tree

QoS Predictions for Non-GBR

Table 4-9 QoS Predictions for Non-GBR

DL Throughput Prediction	
Model/Algorithm	Training Time in Millisec.
Random Forest Regressor	3478.50
K Nearest Neighbour (KNN)	24.70
Decision Tree	33.54
Model Training Time and Total Models	
Total Models	3
Minimum Time (ms)	24.70
Maximum Time (ms)	3478.50
Average Time (ms)	1170.91

NWDAF ATS PerfGo Performance Results

Below are the results from the PerfGo tool after running performance tests on NWDAF ATS:

- Create Subscription
- Update Subscription
- Delete Subscription

Create Subscription**Table 4-10 Create Subscription**

Request Base	Total	Success	Failure	Rate	Latency	Average Rate	Average Latency
ue-mobility-create.create-ue-mob-statistics. [CallFlow]	622	417	194	16.0	734.3ms	9.6	1030.6ms
ue-mobility-create.create-ue-mob-predictions. [CallFlow]	637	430	196	11.0	749.6ms	9.8	996.0ms
abnormal-behaviour-create.create-ab-predictions. [CallFlow]	781	39	731	7.0	1145.5ms	12.1	806.0ms
abnormal-behaviour-create.create-ab-statistics. [CallFlow]	748	34	703	5.0	1201.1ms	11.6	830.4ms

Table 4-10 (Cont.) Create Subscription

Request Base	Total	Success	Failure	Rate	Latency	Average Rate	Average Latency
slice-load-level-create.create-sll-statistics. [CallFlow]	295	32	252	13.0	860.6ms	4.6	2256.8ms
slice-load-level-create.create-sll-predictions. [CallFlow]	304	27	266	14.0	842.8ms	4.7	2177.7ms
nf-load-create.create-nf-load-predictions. [CallFlow]	691	247	434	9.0	813.7ms	10.6	911.7ms
nf-load-create.create-nf-load-statistics. [CallFlow]	705	260	434	11.0	805.9ms	10.9	892.0ms

Update Subscription**Table 4-11 Update Subscription**

Request Base	Total	Success	Failure	Rate	Latency	Average Rate	Average Latency
ue-mobility-update.update-ue-mob. [CallFlow]	80	80	0	3.0	953.4ms	1.4	870.7ms
abnormal-behaviour-update.update-ab. [CallFlow]	80	80	0	3.0	577.1ms	1.5	555.8ms
slice-load-level-update.update-sll. [CallFlow]	70	70	0	2.0	4119.4ms	1.1	1441.4ms
nf-load-update.update-nf-load. [CallFlow]	73	73	0	2.0	610.3ms	1.3	552.4ms

Delete Subscription

Table 4-12 Delete Subscription

Request Base	Total	Success	Failure	Rate	Latency	Average Rate	Average Latency
ue-mobility-delete.delete.e-ue-mob. [CallFlow]	80	80	0	6.0	799.4ms	1.4	611.2ms
abnormal-behaviour-delete.delete.e-ab. [CallFlow]	80	80	0	2.7	577.1ms	1.4	591.0ms
slice-load-level-delete.delete.e-sll. [CallFlow]	65	65	0	2.0	777.3ms	1.2	890.6ms
nf-load-delete.delete.e-nf-load. [CallFlow]	59	59	0	2.0	474.6ms	1.0	536.1ms

User Data Congestion (UDC)**UDC Descriptive Analytics Metrics**

The Subscription service directly calls the Reporting Service to generate reports. The JMeter is used to test the Reporting Service APIs.

URL: <http://nwdaf-cap4c-reporting-service:8080/v1/userdatacongestion/descriptive>

Table 4-13 UDC Descriptive Metrics

Request Type	Threads	Range	Avg. Latency
GET	100	1 hour	432 ms
GET	200	1 hour	434 ms
GET	100	24 hours	604 ms
GET	200	24 hours	1908 ms

UDC Analytics Information Metrics

The Analytic Information service directly calls the Reporting Service to pull the available reports. The JMeter is used to test the Reporting Service API's

URL: <http://nwdaf-cap4c-reporting-service:8080/v1/userdatacongestion/info>

Table 4-14 UDC Analytics Information Metrics

Request Type	Threads	Records in DB	Avg. Latency
GET	100	100 records	516 ms
GET	200	200 records	2229 ms