Oracle® Communications

Cloud Native Unified Data Repository User's Guide

Release 1.6
F31233-02
May 2020

ORACLE

Oracle Communications Cloud Native Unified Data Repository User's Guide, Release 1.6

F31233-02

Copyright © 2019, 2020, 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 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" or “commercial computer software documentation” 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 and Java 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 Overview

Architecture 1-2
Unified Data Repository Features 1-2
My Oracle Support 1-4

2 Configuring User Parameters

3 Using Unified Data Repository (UDR) Console

Logging into CNC Console 3-1
Provisioning - Profile Data 3-3
Provisioning - PCF 3-5
Provisioning - SLF 3-7
Provisioning - UDM 3-9
Provisioning - Schema Management 3-10
Logout of CNC Console 3-12

4 KPIs and Metrics

A Rollback Instructions for PCF Data

What's New in This Guide

This section shares the list of new features introduced in every OCUDR release. For more release specific information, please refer to its release notes.

Release 1.6
The following new features are introduced in this release:

- Supports AM, SM and UEPolicy set as per 29.519 v16.2.0
- HTTPs/TLS support using Ingress API gateway
- CNC-Console integration on provisioning APIs
- Provisioning Gateway integration for SLF provisonining
- Metrics for provisioned data

List of Figures

3-1	CNC Console Login Screen	3-2
3-2	CNC Console Home Page	3-2
3-3	Provisioning - Profile Data	3-3
3-4	Sample Screen: Get - Profile Data - Response	3-4
3-5	Provisioning - PCF	3-5
3-6	Provisioning PCF Get Response	3-6
3-7	Provisioning - SLF	3-7
3-8	Provisioning - SLF - Get Response	3-8
3-9	Provisioning - UDM	3-9
3-10	Provisioning - Schema Management	3-10
3-11	Schema - Get - PCF AM Data	3-11
3-12	CNC Console Logout	3-12

1 Overview

The 5G Unified Data Repository (UDR) is one of the main key component of the 5G Service Based Architecture. UDR is a converged repository, which is used by 5G Network Functions to store the data.

Oracle 5G UDR is implemented as cloud native function and it offers a unified database for storing application, subscription, authentication, service authorization, policy data, session binding and Application state information. It exposes a HTTP2 based RESTful API for NF's and provisioning clients to access the stored data.

Oracle's 5G UDR:

- Leverages a common Oracle Communications Cloud Native Framework
- Is compliant to 3GPP Release 15 specification UDM
- Is compliant to 3GPP v29.519 v16.2 (backward compatible with v15.3.0, by configuration) specification for PCF
- Has tiered architecture providing separation between the connectivity, business logic and data layers
- Uses Oracle MySQL Cluster CGE backend database provides through DB Tier.
- Registers with NRF in the 5G network, so the other NFs in the network can discover UDR through NRF.

As per 3GPP, UDR supports following functionality:

- Storage and retrieval of subscription data by the UDM.
- Storage and retrieval of policy data by the PCF.
- Storage and retrieval of structured data for exposure.
- Storage and retrieval of SLF information, consumed by NRF.
- Application data (including Packet Flow Descriptions (PFDs) for application detection, AF request information for multiple UEs), by the NEF.
- Subscription and Notification feature.

Oracle's 5G UDR provides Unstructured Data Storage Function (UDSF) functionality. This functionality:

- Supports storage and retrieval of unstructured data by any 5G NF. The specifications of UDSF are presently not defined by 3GPP completely.
- This functionality is part of Oracle's 5G UDR solution.

Oracle's 5G UDR provides 5G SLF functionality. This functionality:

- Supports Nudr-groupid-map service as defined by 3GPP
- Complaint with 3GPP Release 16 for APIs to be consumed by 5G NRF
- Supports REST/JSON based provisioning APIs for SLF data

Architecture

The Cloud Native Unified Data Repository architecture has following three tiers:

Connectivity Tier

- Ingress API Gateway (Spring Cloud Gateway [SCG] based) is used as an API gateway that receives all requests and forwards them to the Nudr-drservice service of Business Tier.
- It load balances the traffic and provides required authentication.
- It provides TLS support.

Business Tier

- Provides the business logic of 5G Unified Data Repository.
- It has following three micro services:
  - **nudr-drservice:** The core service that handles flexible URI support, runtime schema validation and connects to Data Tier for DB operations. It provides SLF lookup functionality.
  - **nudr-nrf-client-service:** Handles registration, heartbeat, update and deregistration with Network Repository Function (NRF).
  - **nudr-notify-service:** Handles notification messages to Policy Control Function (PCF) and Unified Data Management (UDM) for data subscriptions.

Data Tier

- Uses Oracle MySQL NDB Cluster, CGE edition as backend database in the DB tier. This provides HA and geo-redundcancy capabilities.

Unified Data Repository Features

This section provides list of the features supported in Oracle Communications Unified Data Repository.

- **Flexible URI support:** A user can define a new URI for any resource at runtime for basic CRUD operations on the resource.
- **Supports multi-keys:** UDR supports multiple keys for a subscriber and provides flexibility to define new ones.
- **Runtime schema validation:** Allows users to modify and validate the schema in use for data storage without service restart.
- **Supports5G PCF's AM**, **SM**, **UEPolicySet** and **UsageMonitoring data APIs**. It is also compliant with 3gpp spec v29.519, v16.2 (backward compatible with v15.3.0). See Rollback to v15.3.0 version of PCF Data
- **Provisioning support via REST/JSON:** UDR provides provisioning APIs for creating subscribers and adding PCF data.
- **Schema versioning:** Using this feature, UDR maintains different versions of schema, starting from default version v0 (software version shipped to customers) and as customer upgrades schema, it creates new versions like v0, v1, v2.
- **Subscription/Notification feature:** The "nudr-drservice" service receives and processes the subscription request to subscribe the subscriber's resources.

Whenever it receives any update or delete request for subscribers (subscribed for notifications), the "nudr-drservice" service sends an internal signal (HTTP2 POST request) to "nudr-notify-service". The "nudr-notify-service" is a new service, which is responsible to send out the notification requests to the target.

- **Cross-reference validation of Sm data:** Using this feature, cross-reference validation of Sm data is done before storing the Sm data in the database. This means that the values of SNSSAI/DNN and LimitIds should be same in "smPolicySnssaiData" and "umData".
- **UDSF API support:** UDR supports basic CRUD operations for UDSF API.
- **UDM API support:** UDR is compliant with 3GPP spec 29.505 v15.4.0. It supports encryption of stored data for certain types of UDM data.
- **Integrated with CNE services:** Like Prometheus/Grafana for metrics, EFK/ Kibana for logging and Jaeger for tracing
- **Flexible URI support:** UDR allows you to define a new URI for any resource at runtime for basic CRUD operations on the resource.
- **Supports multi-keys:** Supports multiple keys for a subscriber and provides flexibility to define new ones.
- **Runtime schema validation:** UDR allows you to modify and validate schema for the data stored without any service restart.
- Utilizes the DB Tier (MySQL NDB Cluster)'s encryption technology
- **5G SLF functionality:** UDR provides Nudr-groupid-map service towards 5G NRF. It is compliant with 3GPP release 16, spec 29.504 v16.2.0
- Supports TLS with Ingress API gateway.
- Integrated with ProvGwy for receiving provisioning updates to UDR. See ProvGwy documentation
- Integrated with CNC-C for manual subscriber provisioning on the GUI.

List of Operations Supported

Unified Data Repository supports the following operations.

Subscriber Profile Related Operations for Provisioning

- **Create Subscriber**: Provisions a subscriber with the given set of keys.
- **Update Subscriber**: Updates the subscriber information.
- **Get subscriber**: Retrieves the subscriber information completely.
- **Delete Subscriber**: Deletes the subscriber information and related data.
- **Subscription Request**: Subscribes to a subscriber's resources and get notified in case of updates.
- **Notification**: UDR generates notification when there are updates to subscriber's resources, subscribed in previous request. These notifications are sent to notification URI received in subscription request.

NF Data Related Operations as per 3GPP

- **Insert Data**: Creates a subscriber if not present and inserts the specific data (policy_data/udsf_data/udm_data/SLF Data).

- **Update Data**: Creates a subscriber if not present and updates the complete data for a particular subscriber as given in request.
- **Patch Data**: Supports patch update of a specific data or parts of it. If not present, it adds the data. This operation is valid only when subscriber is already present.
- **Get Data**: Retrieves the requested data for a particular subscriber.
- **Delete Data**: Deletes the requested data for a particular subscriber.

My Oracle Support

My Oracle Support 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 MOS, 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.

2
Configuring User Parameters

The UDR micro services have configuration options. The user should be able to configure them via deployment values.yaml.

> **Note:**
>
> The default value of some of the settings may change.

> **Note:**
>
> - **NAME**: is the release name used in helm install command
> - **NAMESPACE**: is the namespace used in helm install command
> - **K8S_DOMAIN**: is the default kubernetes domain (svc.cluster.local)

Default Helm Release Name:- ocudr

Following table provides the parameters for **global configurations**.

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
dockerRegistry	Docker registry from where the images will be pulled	reg-1:5000	Not applicable	
mysql.dbService Name	DB service to connect	mysql-connectivity-service.occne-infra	Not applicable	This is a CNE service used for db connection. Default name used on CNE is the same as configured.
mysql.port	Port of DBService Connection	3306	Not applicable	
jaeger.service.name	Jaegar Service Name installed in CNE	occne-tracer-jaeger-collector.occne-infra	Not applicable	
jaeger.service.port	Jaegar Service Port installed in CNE	9411	Not applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
hikari.poolsize	Connection pool size	25	Not Applicable	The hikari pool connection size to be created at start up
dbenc.shavalue	Encryption Key size	256	256 or 512	
serviceAccountName	Service account name	null	Not Applicable	The serviceaccount, role and rolebindings required for deployment should be done prior installation. Use the created serviceaccountname here.
prefix.container	Container configurable prefix	null	Not Applicable	If this is configured with some value, the same will be used as prefix for container names on different pods of UDR deployment. If not configured, release name will be used as preifx.
prefix.configmap	Configmap configurabe prefix	null	Not Applicable	If this is configured with some value, the same will be used as prefix for configmap names. If not configured, release name will be used as preifx.
prefix.hpa	HPA configurable prefix	null	Not Applicable	If this is configured with some value, the same will be used as prefix for HPA names. If not configured, release name will be used as preifx.

Following table provides the parameters for **nudr-drservice micro service**.

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
image.name	Docker Image name	ocudr/ nudr_datarepository_service	Not applicable	
image.tag	Tag of Image	1.6.0	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
subscriber.autocreate	Flag to enable auto creation of subscriber	true	true/false	This flag will enable auto creation of subscriber when creating data for a non existent subscriber.
validate.smdata	Flag to enable correlation feature for smdata	false	true/false	This flag will control the correlation feature for smdata. This flag must be false if using v16.2.0 for PCF data.
logging.level.root	Log Level	WARN	Possible Values - WARN INFO DEBUG	Log level of the nudr-drservice pod
deployment.replicaCount	Replicas of nudr-drservice pod	2	Not applicable	Number of nudr-drservice pods to be maintained by replica set created with deployment
minReplicas	Minimum Replicas	2	Not applicable	Minimum number of pods
maxReplicas	Maximum Replicas	4	Not applicable	Maximum number of pods
service.http2enabled	Enabled HTTP2 support flag for rest server	true	true/false	Enable/Disable HTTP2 support for rest server
service.type	UDR service type	ClusterIP	Possbile Values- ClusterIP NodePort LoadBalancer	The kubernetes service type for exposing UDR deployment Note: Suggested to be set as ClusterIP (default value) always

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
service.port.http	HTTP port	5001	Not applicable	The http port to be used in nudr-drservice service
service.port.https	HTTPS port	5002	Not applicable	The https port to be used for nudr-drservice service
service.port.management	Management port	9000	Not applicable	The actuator management port to be used for nudr-drservice service
resources.requests.cpu	Cpu Allotment for nudr-drservice pod	3	Not applicable	The cpu to be allocated for nudr-drservice pod during deployment
resources.requests.memory	Memory allotment for nudr-drservice pod	4Gi	Not applicable	The memory to be allocated for nudr-drservice pod during deployment
resources.limits.cpu	Cpu allotment limitation	3	Not applicable	
resources.limits.memory	Memory allotment limitation	4Gi	Not applicable	
resources.target.averageCpuUtil	CPU utilization limit for autoscaling	80	Not Applicable	CPU utilization limit for creating HPA
notify.port.http	HTTP port on which notify service is running	5001	Not applicable	
notify.port.https	HTTPS port on which notify service is running	5002	Not applicable	

Following table provides the parameters for **nudr-notify-service micro service**.

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
enabled	flag for enabling or disabling nudr-notify-service	true	true or false	For SLF deployment, this micro service must be disabled.
image.name	Docker Image name	ocudr/nudr_notify_service	Not applicable	
image.tag	Tag of Image	1.6.0	Not applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
notification.retrycount	Number of notifications to be attempted	3	Range: 1 - 10	Number of notification attempts to be done in case of notification failures. Whether retry should be done will be based on notification.retryerrorcodes configuration.
notification.retryinterval		5	Range: 1 - 60 Unit: Seconds	The retry interval for notifications in case of failure. Unit is in seconds. Whether retry should be done will be based on notification.retryerrorcodes configuration.
notification.retryerrorcodes	Notification failures eligible for retry	"400,429,500,503"	Valid HTTP status codes comma seperated	Comma separated error code should be given. These error codes will be eligible for retry notifications in case of failures.
logging.level.root	Log Level	WARN	Possible Values - WARN INFO DEBUG	Log level of the notify service pod
deployment.replicaCount	Replicas of nudr-notify-service pod	2	Not applicable	Number of nudr-notify-service pods to be maintained by replica set created with deployment
minReplicas	Minimum Replicas	2	Not applicable	Minimum number of pods

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
maxReplicas	Maximum Replicas	4	Not applicable	Maximum number of pods
service.http2enabled	Enabled HTTP2 support flag	true	true/false	This is a read only parameter. Do not change this value
service.type	UDR service type	ClusterIP	Possbile Values- ClusterIP NodePort LoadBalancer	The kubernetes service type for exposing UDR deployment Note: Suggested to be set as ClusterIP (default value) always
service.port.http	HTTP port	5001	Not applicable	The http port to be used in notify service to receive signals from nudr-notify-service pod.
service.port.https	HTTPS port	5002	Not applicable	The https port to be used in notify service to receive signals from nudr-notify-service pod.
service.port.management	Management port	9000	Not applicable	The actuator management port to be used for notify service.
resources.requests.cpu	Cpu Allotment for nudr-notify-service pod	3	Not applicable	The cpu to be allocated for notify service pod during deployment
resources.requests.memory	Memory allotment for nudr-notify-service pod	4Gi	Not applicable	The memory to be allocated for nudr-notify-service pod during deployment
resources.limits.cpu	Cpu allotment limitation	3	Not applicable	
resources.limits.memory	Memory allotment limitation	4Gi	Not applicable	
resources.target.averageCpuUtil	CPU utilization limit for autoscaling	80	Not Applicable	CPU utilization limit for creating HPA

Following table provides the parameters for **nudr-nrf-client-service micro service**.

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
enabled	flag for enabling or disabling nudr-nrf-client-service	true	true/false	
host.baseurl	NRF url for registration	http://ocnrf-ingressgateway.mynrf.svc.cluster.local/nnrf-nfm/v1/nf-instances	Not applicable	Url used for udr to connect and register with NRF
host.proxy	Proxy Setting	NULL	nrfClient.host	Proxy setting if required to connect to NRF
ssl	SSL flag	false	true/false	SSL flag to enable SSL with udr nrf client pod
logging.level.root	Log Level	WARN	Possible Values - WARN INFO DEBUG	Log level of the UDR nrf client pod
image.name	Docker Image name	ocudr/nudr_nrf_client_service	Not applicable	
image.tag	Tag of Image	1.6.0	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
heartBeatTimer	Heart beat timer	90	Unit: Seconds	
groupId	Group ID of UDR	udr-1	Not applicable	
capacityMultiplier	Capacity of UDR	500	Not applicable	Capacity multiplier of UDR based on number of UDR pods running
supirange	Supi Range supported with UDR	[{\"start\": \"10000000000\", \"end\": \"20000000000\"}]	Valid start and end supi range	
priority	Priority	10	Priority to be sent in registration request	Priority to be sent in registration request

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
livenessProbeUrl	Liveness probe URL for nudr-drservice/api-gateway	http://nudr-notify-service.myudr.svc.cluster.local:9000/actuator/health http://nudr-drservice.myudr.svc.cluster.local:9000/actuator/health	Not Applicable	URL used by nrf-client-service to check liveness probe of nudr-drservice, nudr-notify-service and ocudr-ambassador pods. **Note:** Be cautious in updating this value. Should consider helm release name, namespace used for udr deployment and name resolution setting in k8s.
fqdn	UDR FQDN	ocudr-ingressgateway.myudr.svc.cluster.local	Not Applicable	FQDN to used for registering in NRF for other NFs to connect to UDR. Note: Be cautious in updating this value. Should consider helm release name, namespace used for udr deployment and name resolution setting in k8s.
gpsirange	Gpsi Range supported with UDR	[{\"start\": \"10000000000\", \"end\": \"20000000000\"}]	Valid start and end gpsi range	
endpointLabelSelector	Pod name of ingress gateway	ocudr-ingressgateway	This should be changed based on the name space that you created.	End Point Label Selector is used to get the port number of the running ingress gateway pod that is deployed.
masterIp	Master IP of which we deployed	10.0.0.0	This should be changed with the master ip which we deployed	Master IP is used to send the ipv4 address to the nrf while registration.

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
plmnvalues	Plmn values range that it supports	[{\"mnc\": \"14\", \"mcc\": \"310\"}]	This values can be changed that the range it supports	Plmn values are sent to nrf during regisration from UDR.
scheme	scheme in which udr supports	http	This can be changed to https.	scheme which we send to NRF during registration
resources.requests.cpu	Cpu Allotment for nudr-notify-service pod	1	Not applicable	The cpu to be allocated for nrf client service pod during deployment
resources.requests.memory	Memory allotment for nudr-notify-service pod	2Gi	Not applicable	The memory to be allocated for nrf client service pod during deployment
resources.limits.cpu	Cpu allotment limitation	1	Not applicable	
resources.limits.memory	Memory allotment limitation	2Gi	Not applicable	

Following table provides parameters for **ocudr-ingressgateway micro service (API Gateway)**

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
type	ocudr-ingressgateway service type	LoadBalancer	Possbile Values- ClusterIP NodePort LoadBalancer	
metalLbIpAllocationEnabled	Enable or disable Address Pool for Metallb	true	true/false	
metalLbIpAllocationAnnotation	Address Pool for Metallb	metallb.universe.tf/address-pool: signaling	Not applicable	
staticNodePortEnabled	If Static node port needs to be set, then set staticNodePortEnabled flag to true and provide value for staticNodePort	false	Not applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
staticHttpNodePort	static http node port value need to be provided	30075	can be changed based of user requirement.	
staticHttpsNodePort	static https node port value need to be provided	30043	can be changed based of user requirement.	
image.name	Docker image name	ocudr/ocingress_gateway	Not applicable	
image.tag	Image version tag	1.6.2	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
initContainersImage.name	Docker Image name	ocudr/configurationinit	Not applicable	
initContainersImage.tag	Image version tag	1.1.1	Not applicable	
initContainersImage.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
updateContainersImage.name	Docker Image name	ocudr/configurationupdate	Not applicable	
updateContainersImage.tag	Image version tag	1.1.1	Not applicable	
updateContainersImage.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
service.ssl.privateKey.k8SecretName	name of the secret which stores keys and certificates	ocudr-gateway-secret	Not applicable	
service.ssl.privateKey.k8NameSpace	namespace in which secret is created	ocudr	Not applicable	
service.ssl.privateKey.rsa.fileName	rsa private key stored in the secret	rsa_private_key_pkcs1.pem	Not applicable	
service.ssl.privateKey.ecdsa.fileName	ecdsa private key stored in the secret	ecdsa_private_key_pkcs8.pem	Not applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
service.ssl.certificate.k8SecretName	name of the secret which stores keys and certificates	ocudr-gateway-secret	Not applicable	
service.ssl.certificate.k8NameSpace	namespace in which secret is created	ocudr	Not applicable	
service.ssl.certificate.rsa.fileName	rsa certificate stored in the secret	apigatewayrsa.cer	Not applicable	
service.ssl.certificate.ecdsa.fileName	ecdsa certificate stored in the secret	apigatewayecdsa.cer	Not applicable	
service.ssl.caBundle.k8SecretName	name of the secret which stores keys and certificates	ocudr-gateway-secret	Not applicable	
service.ssl.caBundle.k8NameSpace	namespace in which secret is created	ocudr	Not applicable	
service.ssl.caBundle.fileName	ca Bundle stored in the secret	caroot.cer	Not applicable	
service.ssl.keyStorePassword.k8SecretName	name of the secret which stores keys and certificates	ocudr-gateway-secret	Not applicable	
service.ssl.keyStorePassword.k8NameSpace	namespace in which secret is created	ocudr	Not applicable	
service.ssl.keyStorePassword.fileName	keyStore password stored in the secret	key.txt	Not applicable	
service.ssl.trustStorePassword.k8SecretName	name of the secret which stores keys and certificates	ocudr-gateway-secret	Not applicable	
service.ssl.trustStorePassword.k8NameSpace	namespace in which secret is created	ocudr	Not applicable	
service.ssl.trustStorePassword.fileName	trustStore password stored in the secret	trust.txt	Not applicable	
resources.limits.cpu	Cpu allotment limitation	3	Not applicable	
resources.limits.memory	Memory allotment limitation	4Gi	Not applicable	
resources.requests.cpu	Cpu allotment for ocudr-endpoint pod	3	Not Applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
resources.requests.memory	Memory allotment for ocudr-endpoint pod	4Gi	Not Applicable	
resources.target.averageCpuUtil	CPU utilization limit for autoscaling	80	Not Applicable	
minAvailable	Number of pods always running	2	Not Applicable	
minReplicas	Min replicas to scale to maintain an average CPU utilization	2	Not applicable	
maxReplicas	Max replicas to scale to maintain an average CPU utilization	5	Not applicable	
log.level.root	Logs to be shown on ocudr-endpoint pod	WARN	valid level	
log.level.ingress	Logs to be shown on ocudr-ingressgateway pod for ingress related flows	INFO	valid level	
log.level.oauth	Logs to be shown on ocudr-ingressgateway pod for oauth related flows	INFO	valid level	
fullnameOverride	Name to be used for deployment	ocudr-ingressgateway	Not applicable	This config is commented by default.
initssl	To Initialize SSL related infrastructure in init/update container	false	Not Applicable	
jaegerTracingEnabled	Enable/Disable Jaeger Tracing	false	true/false	
openTracing.jaeger.udpSender.host	Jaeger agent service FQDN	jaeger-agent.cne-infra	Valid FQDN	
openTracing.jaeger.udpSender.port	Jaeger agent service UDP port	6831	Valid Port	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
openTracing.jaeger.probabilisticSampler	Probablistic Sampler on Jaeger	0.5	Range: 0.0 - 1.0	Sampler makes a random sampling decision with the probability of sampling. For example, if the value set is 0.1, approximately 1 in 10 traces will be sampled
oauthValidatorEnabled	OAUTH Configuration	false	Not Applicable	
enableIncomingHttp	Enabling for accepting http requests	true	Not Applicable	
enableIncomingHttps	Enabling for accepting https requests	false	true or false	
enableOutgoingHttps	Enabling for sending https requests	false	true or false	
maxRequestsQueuedPerDestination	Queue Size at the ocudr-endpoint pod	5000	Not Applicable	
maxConnectionsPerIp	Connections from endpoint to other microServices	10	Not Applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
routesConfig	Routes configured to connect to different micro services of UDR	- id:traffic_mapping_http uri: http://{{ .Release.Name }}-nudr-drservice:5001 path: /nudr-dr/** - id: traffic_mapping_http_prov uri: http://{{ .Release.Name }}-nudr-drservice:5001 path: /nudr-dr-prov/** - id: traffic_mapping_http_mgmt uri: http://{{ .Release.Name }}-nudr-drservice:5001 path: /nudr-dr-mgm/** - id: traffic_mapping_http_udsf uri: http://{{ .Release.Name }}-nudr-drservice:5001 path: /nudsf-dr/** - id: traffic_mapping_http_group uri: http://{{ .Release.Name }}-nudr-drservice:5001 path: /nudr-group-id-map/** - id: traffic_mapping_http_group_prov uri: http://{{ .Release.Name }}-nudr-	Not Applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
		drservice: 5001 path: / nudr-group-id-map-prov/**		

3 Using Unified Data Repository (UDR) Console

In this chapter, you will learn to login to CNC Console application and use UDR for performing provisioning operations for profile-data, PCF, SLF, UDM and schema management.

> **Note:**
>
> As on UDR 1.6 and CNCC-1.1, only provisioning operations are supported. Configuration updates are not yet supported.

Logging into CNC Console

A user can use UDR integrated with CNC Console only after logging successfully into the CNC Console application. In order to login successfully to the CNC Console, the user needs to make the following updates to the hosts file available at the **C:\Windows\System32\drivers\etc** location.

In the Windows system, user needs to open the **hosts** file in the notepad as an Administrator and append the following set of lines at the end:

Example:

```
10.75.224.58 cncc-iam-ingress-gateway.cncc.svc.cluster.local
10.75.224.58 cncc-core-ingress-gateway.cncc.svc.cluster.local
```

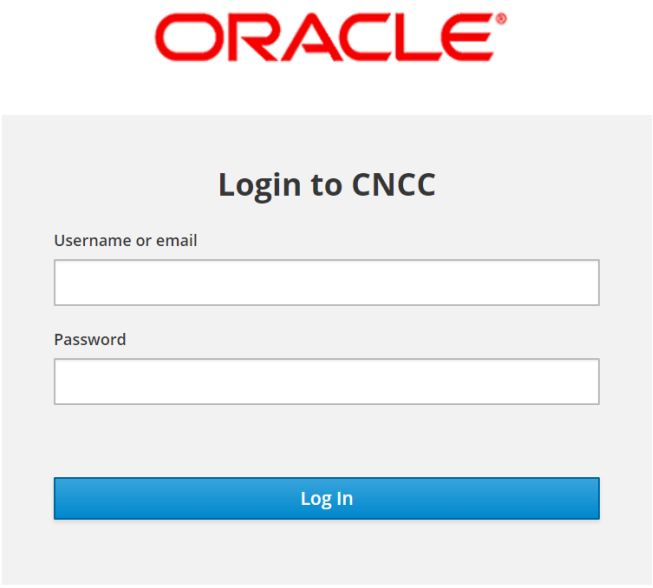
> **Note:**
>
> The IP Address in the above lines may change when deloyment cluster changes.

Save and close the notepad.

To login to CNC Console:

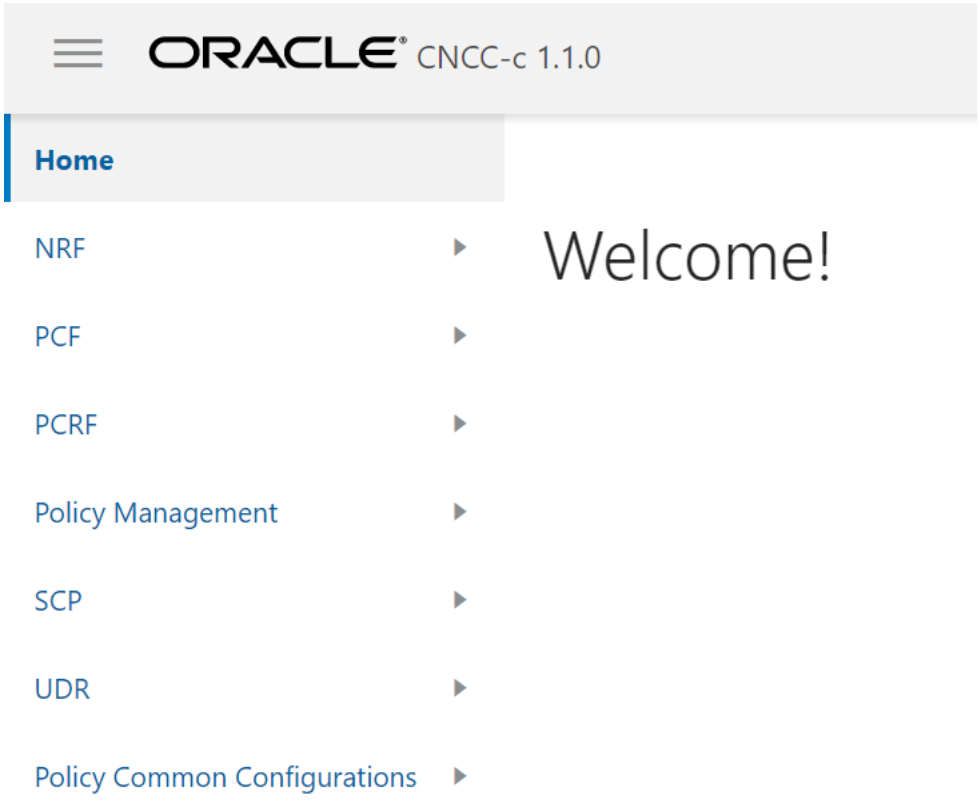
1. Type **http://cncc-core-ingress-gateway.cncc.svc.cluster.local:<PortNumber>/** in the web browser and press Enter. Following screen appears:

Figure 3-1 CNC Console Login Screen



2. Enter the **Username** and **Password** and click **Log In**. Following screen appears:

Figure 3-2 CNC Console Home Page



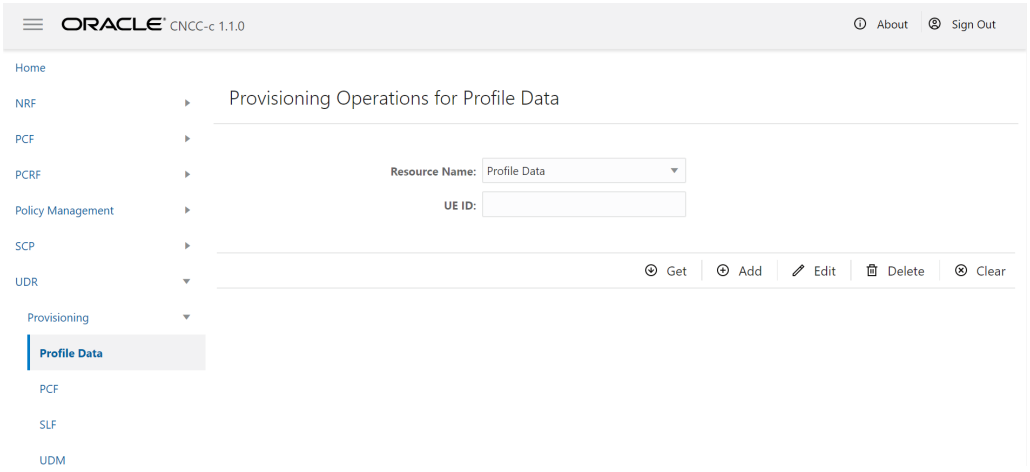
This is the CNC Console Home Page from where a user can navigate to different NF services. To use UDR services integrated with CNC Console, click **UDR** in the left navigation pane.

Provisioning - Profile Data

In this section, you will learn to operate provisioning operations for Profile Data.

In the CNC Console application, click **UDR** → **Provisioning** → **Profile Data** in the left navigation pane. Following screen appears:

Figure 3-3 Provisioning - Profile Data



In the Provisioning Operations for Profile Data screen, user can:

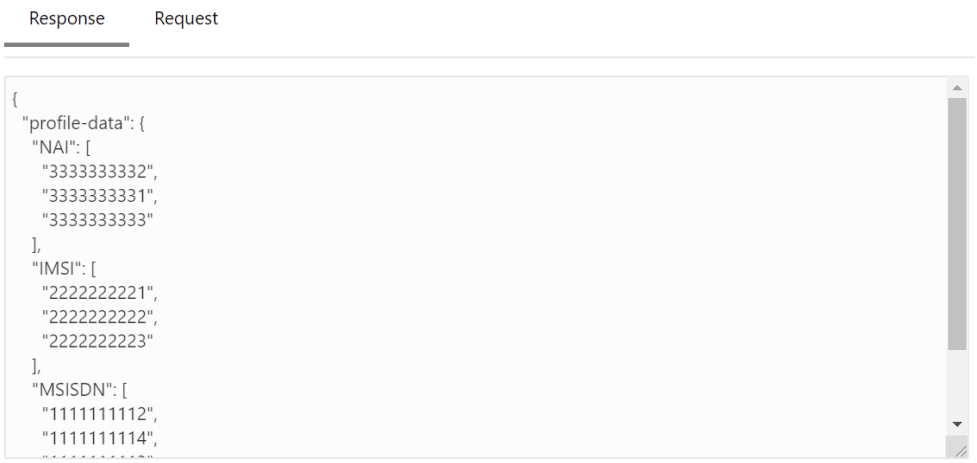
- Extract profile data from UDR database on the basis of UE ID entered.
- Add new profile data to the UDR database
- Modify an existing profile data on the basis of UE ID
- Delete an existing profile data that is no more needed in the UDR database

Extracting Profile Data

To extract a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Get**. A confirmation message, '**Fetched Successfully**' appears for a second and the result appears in the **Response** text area as follows:

Figure 3-4 Sample Screen: Get - Profile Data - Response



4. An **Error message summary** appears in case of incorrect UE ID entered.

Adding a Profile Data

To add a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Add**. The **Request** text area appears.
4. Type-in the profile data that you want to add to the UDR database.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

Editing a Profile Data

To edit a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Edit**. The **Request** text area displays an existing profile data.
4. Type-in the updated profile data.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

Deleting a Profile Data

To delete a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Delete**.
4. The '**Deleted Successfully**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

Clearing a Profile Data

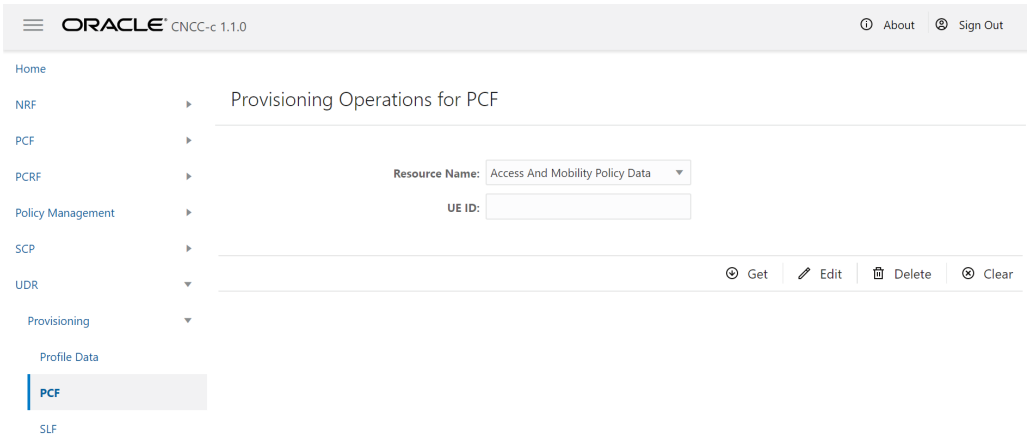
Click the **Clear** option to clear all the text fields and text areas of the Provisioning Operations for Profile Data.

Provisioning - PCF

In this section, you will learn to operate provisioning operations for PCF (Policy Control Function).

In the CNC Console application, click **UDR** → **Provisioning** → **PCF** in the left navigation pane. The following screen appears:

Figure 3-5 Provisioning - PCF



In the Provisioning Operations for PCF screen, user can:

- Select **Resource Name** as either Access And Mobility Policy Data, Session Management Policy Data or UE Policy Set.
- Extract PCF data from UDR database on the basis of UE ID entered
- Modify an existing PCF data on the basis of UE ID

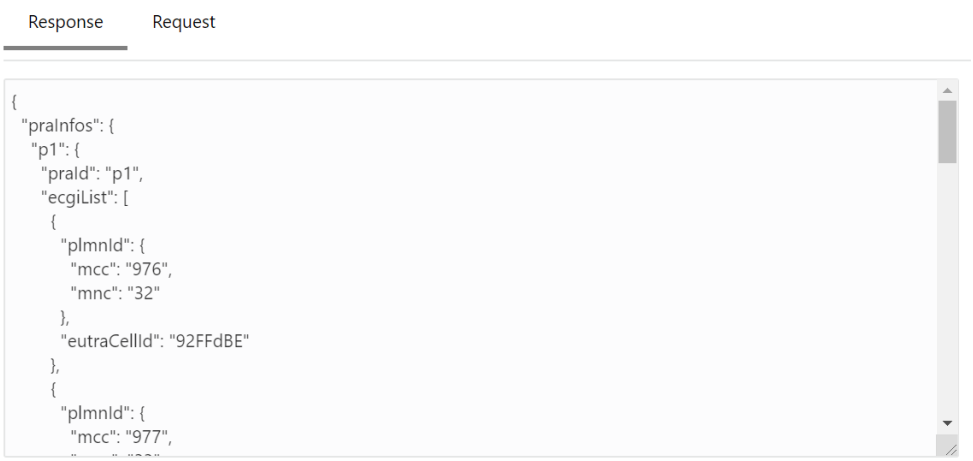
- Delete an existing PCF data that is no more needed in the UDR database

Extracting PCF Data

To extract the PCF data:

1. Select **Resource Name** as either Access And Mobility Policy Data, Session Management Policy Data or UE Policy Set.
2. Type an appropriate **UE ID**. For example: msisdn-9111111112
3. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area as follows:

Figure 3-6 Provisioning PCF Get Response



4. An **Error message summary** appears in case of incorrect UE ID entered.

Editing a PCF Data

To edit a PCF data:

1. Select the **Resource Name** as either Access And Mobility Policy Data, Session Management Policy Data or UE Policy Set.
2. Type an appropriate **UE ID**. For example: msisdn-9111111112
3. Click **Edit**. The **Request** text area displays an existing PCF data.
4. Type-in the updated profile data.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

Deleting a PCF Data

To delete a PCF Data:

1. Select the **Resource Name** as either Access And Mobility Policy Data, Session Management Policy Data or UE Policy Set.

2. Type an appropriate **UE ID**. For example: msisdn-9111111112
3. Click **Delete**.
4. The '**Deleted Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

Clearing a Profile Data

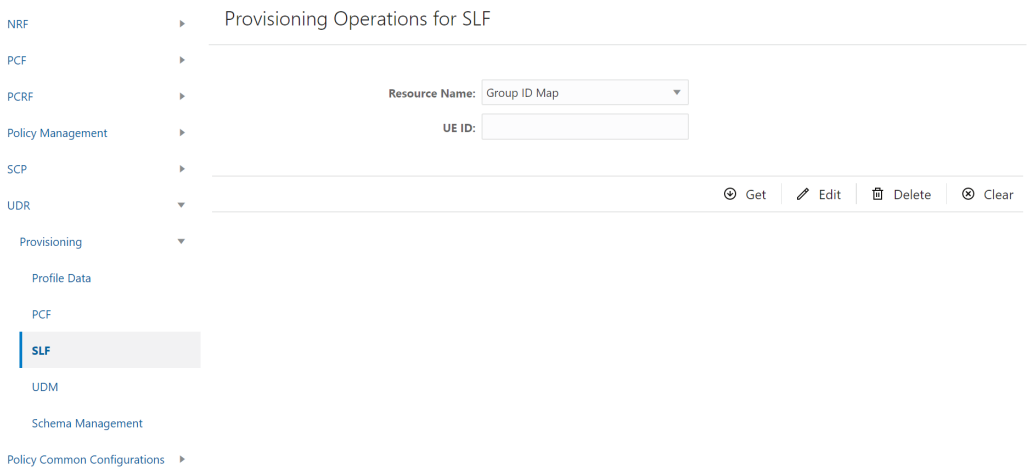
Click the **Clear** option to clear the text fields and text areas of the Provisioning Operations for PCF screen.

Provisioning - SLF

In this section, you will learn to operate provisioning operations for SLF (Subscriber Location Function).

In the CNC Console application, click **UDR → Provisioning → SLF** in the left navigation pane. The following screen appears:

Figure 3-7 Provisioning - SLF



In the Provisioning Operations for SLF screen, user can:

- Extract SLF data from UDR database on the basis of UE ID entered
- Modify an existing SLF data on the basis of UE ID
- Delete an existing SLF data that is no more needed in the UDR database

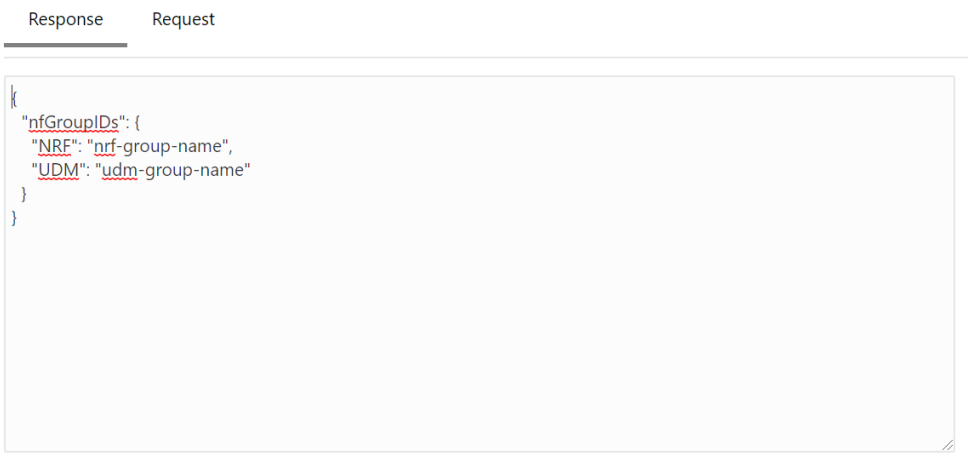
Extracting SLF Data

To extract the SLF data:

1. Ensure that the **Resource Name** is set as Group ID Map.

2. Type an appropriate **UE ID**. For example: msisdn-1111111112
3. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area as follows:

Figure 3-8 Provisioning - SLF - Get Response



4. An **Error message summary** appears in case of incorrect UE ID entered.

Editing a SLF Data

To edit a SLF data:

1. Ensure that the **Resource Name** is set as 'Group ID Map'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111112
3. Click **Edit**. The **Request** text area displays an existing SLF data.
4. Type-in the updated SLF data.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
> If a user enters any incorrect input, an Error message summary appears.

Deleting a SLF Data

To delete a SLF Data:

1. Ensure that the **Resource Name** is set as 'Group ID Map'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111112
3. Click **Delete**.
4. The '**Deleted Successfully.**' message appears.

> **Note:**
> If a user enters any incorrect input, an Error message summary appears.

Clearing a Profile Data

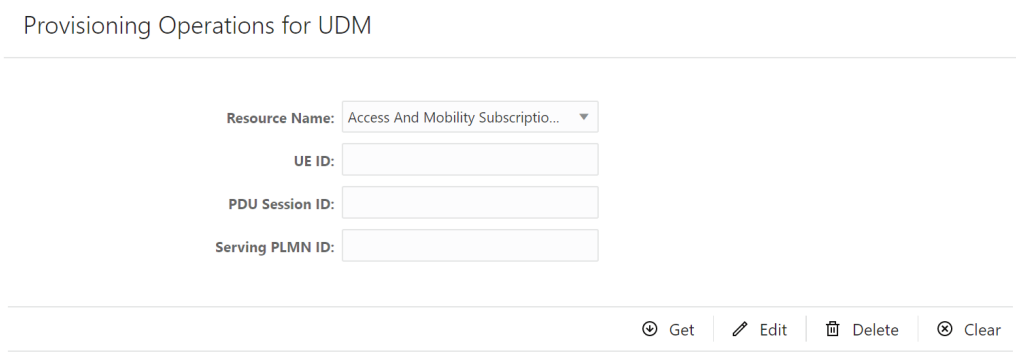
Click the **Clear** option to clear the text fields and text areas of the Provisioning Operations for SLF screen.

Provisioning - UDM

In this section, you will learn to operate provisioning operations for UDM (Unified Data Management).

In the CNC Console application, click **UDR** → **Provisioning** → **UDM** in the left navigation pane. The following screen appears:

Figure 3-9 Provisioning - UDM



In the Provisioning Operations for UDM screen, user can extract, modify and delete UDM data on the basis of Resource Name, UE ID, PDU Session ID and Serving PLMN ID entered.

Extracting UDM Data

To extract the UDM data:

1. Select the **Resource Name** from the drop-down list.
2. Type an appropriate **UE ID**, **PDU Session ID** and **Serving PLMN ID** in their respective fields.
3. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area.
4. An **Error message summary** appears in case of incorrect UE ID entered.

Editing a UDM Data

To edit a UDM data:

1. Select the **Resource Name**.
2. Type an appropriate **UE ID**, **PDU Session ID** and **Serving PLMN ID** in their respective fields.
3. Click **Edit**. The **Request** text area displays an existing UDM data.

4. Edit as required and click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

Deleting a UDM Data

To delete a UDM Data:

1. Enter the **Resource Name**.
2. Type an appropriate **UE ID**, **PDU Session ID** and **Serving PLMN ID** in their respective fields.
3. Click **Delete**.
4. The '**Deleted Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

Clearing a Profile Data

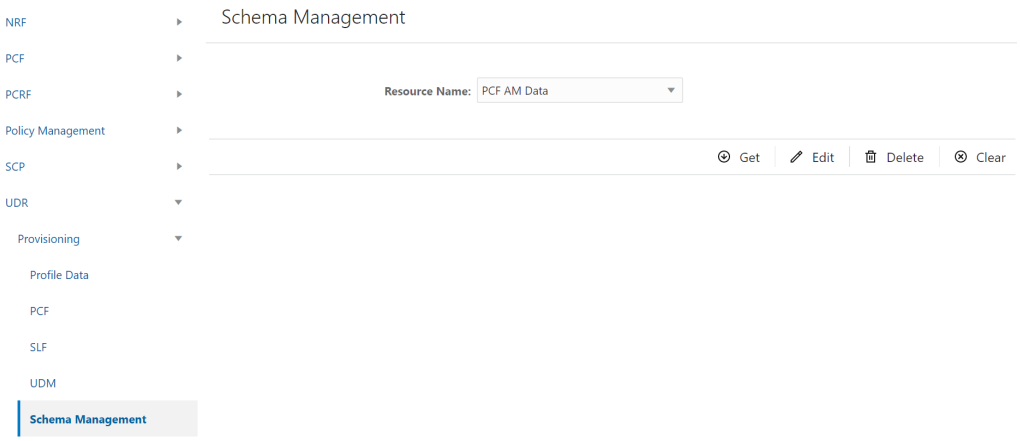
Click the **Clear** option to clear the text fields and text areas of the Provisioning Operations for UDM screen.

Provisioning - Schema Management

In this section, you will learn to operate provisioning operations for schemas.

In the CNC Console application, click **UDR → Provisioning → Schema Management** in the left navigation pane. The following screen appears:

Figure 3-10 Provisioning - Schema Management



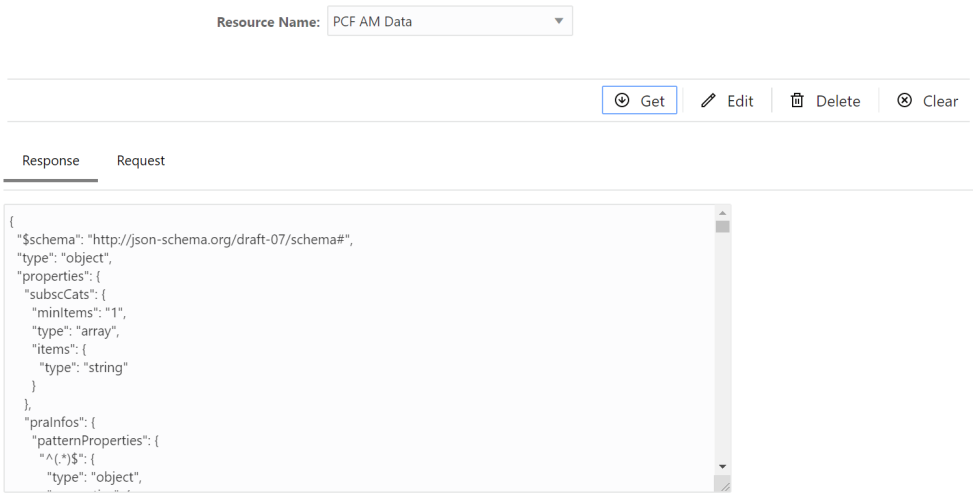
In the **Schema Management** screen, user can extract, modify and delete a schema on the basis of Resource Name. This resource name can be either PCF AM Data, PCF SM Data or PCF UE Policy Set.

Extracting a Schema

To extract a schema:

1. Select the **Resource Name**. It can be either PCF AM Data, PCF SM Data or PCF UE Policy Set.
2. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area as follows:

Figure 3-11 Schema - Get - PCF AM Data



3. An **Error message summary** appears in case of incorrect UE ID entered.

Editing a Schema

To edit a schema:

1. Select the **Resource Name**.
2. Click **Edit**. The **Request** text area displays the selected schema.
3. Edit as required and click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user edits the schema inappropriately, an Error message summary appears.

Deleting a Schema

To delete a schema:

1. Select the **Resource Name**
2. Click **Delete**.

3. The '**Deleted Successfully.**' message appears.

> **Note:**
>
> If a user selects a schema that does not exists, an Error message summary appears.

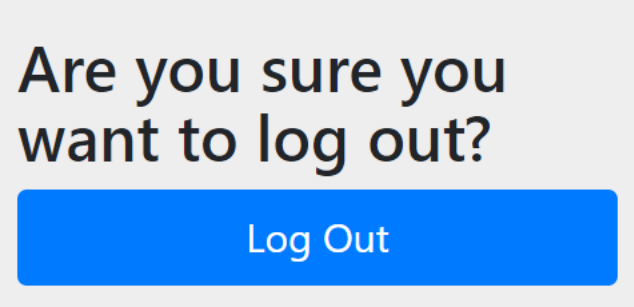
Clearing a Profile Data

Click the **Clear** option to clear the text fields and text areas of the Schema Management screen.

Logout of CNC Console

To logout of the CNC Console application, click the **Sign Out** link available at the top right corner. The following confirmation box appears:

Figure 3-12 CNC Console Logout



Click **Log Out**. The user exits the application and the CNC Console login screen appears.

4
KPIs and Metrics

This section provides information about the KPIs and Metrics for Oracle Communications Unified Data Repository (OCUDR).

> **Note:**
>
> A sample Grafana json is available in the UDR Customer Documentation on Oracle Help Center at . You can download and import the file to your repository. http://docs.oracle.com/en/industries/communications/cloud-native-core/release21.html

Metrics

Following metrics are applicable to **Ingress API Gateway**.

Metric Details	UDR Microservice	Service Operation	Response Code	Notes
oc_ingressgateway_http_requests_total	ocudr-ingress-gateway	All	Not Applicable	oc_ingressgateway_http_requests_total
oc_ingressgateway_http_responses_total	ocudr-ingress-gateway	All	Not Applicable	oc_ingressgateway_http_responses_total
Total no of POST requests received by UDR	ocudr-ingress-gateway	PostRequests	201	oc_ingressgateway_http_requests_total{Method="POST"}
Total no of PUT requests received by UDR	ocudr-ingress-gateway	PutRequests	201	oc_ingressgateway_http_requests_total{Method="PUT"}
Total no of GET requests received by UDR	ocudr-ingress-gateway	GetRequests	200	oc_ingressgateway_http_requests_total{Method="GET"}
Total no of DELETE requests received by UDR	ocudr-ingress-gateway	DeleteRequests	204	oc_ingressgateway_http_requests_total{Method="DELETE"}
Total no of PATCH requests received by UDR	ocudr-ingress-gateway	PatchRequests	204	oc_ingressgateway_http_requests_total{Method="PATCH"}

Metric Details	UDR Microservice	Service Operation	Response Code	Notes
Total no of success POST response with status as 2xx	ocudr-ingress-gateway	PostRequests	2xx	oc_ingressgateway_http_responses_total{Method="POST",Status=~"2.*"}
Total no of success PUT response with status as 2xx	ocudr-ingress-gateway	PutRequests	2xx	oc_ingressgateway_http_responses_total{Method="PUT",Status=~"2.*"}
Total no of success GET response with status as 2xx	ocudr-ingress-gateway	GetRequests	2xx	oc_ingressgateway_http_responses_total{Method="GET",Status=~"2.*"}
Total no of success DELETE response with status as 2xx	ocudr-ingress-gateway	DeleteRequests	2xx	oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"2.*"}
Total no of success PATCH response with status as 2xx	ocudr-ingress-gateway	PatchRequests	2xx	oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"2.*"}
Total no of failure POST response with status as 4xx	ocudr-ingress-gateway	PostRequests	4xx	oc_ingressgateway_http_responses_total{Method="POST",Status=~"4.*"}
Total no of failure PUT response with status as 4xx	ocudr-ingress-gateway	PutRequests	4xx	oc_ingressgateway_http_responses_total{Method="PUT",Status=~"4.*"}
Total no of failure GET response with status as 4xx	ocudr-ingress-gateway	GetRequests	4xx	oc_ingressgateway_http_responses_total{Method="GET",Status=~"4.*"}
Total no of failure DELETE response with status as 4xx	ocudr-ingress-gateway	DeleteRequests	4xx	oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"4.*"}
Total no of failure PATCH response with status as 4xx	ocudr-ingress-gateway	PatchRequests	4xx	oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"4.*"}

Following metrics are applicable to the **nudr-dr service**.

Category	Sub-category	Description	Metric Name	Notes
Rest Controller UDR	Aggregate	UDRTotalRequests	udr_rest_allrequests_total	Total number of requests received by nudr-drservice (All UDR operations)
		UDRTotalResponses	udr_rest_allresponse_total	Total number of responses sent by nudr-drservice (All UDR operations)
		UDRTotalSuccessResponses	udr_rest_successResponse_total	Total number of success responses sent by nudr-drservice (All UDR operations)
		UDRTotalFailureResponses	udr_rest_failureResponse_total	Total number of failure responses sent by nudr-drservice (All UDR operations)
	PUT	UDRPutRequests	udr_rest_request_total{Method="PUT"}	Total number of PUT requests received by nudr-drservice (UDR related)
		UDRPutSuccessResponses	udr_rest_response_success_total{Method="PUT"}	Total number of PUT success responses sent by nudr-drservice (UDR related)
		UDRPutFailureResponses	udr_rest_response_failure_total{Method="PUT"}	Total number of PUT failure responses sent by nudr-drservice (UDR related)
	PATCH	UDRPatchRequests	udr_rest_request_total{Method="PATCH"}	Total number of PATCH requests received by nudr-drservice (UDR related)
		UDRPatchSuccessResponses	udr_rest_response_success_total{Method="PATCH"}	Total number of PATCH success responses sent by nudr-drservice (UDR related)
		UDRPatchFailureResponses	udr_rest_response_failure_total{Method="PATCH"}	Total number of PATCH failure responses sent by nudr-drservice (UDR related)

Category	Sub-category	Description	Metric Name	Notes
	GET	UDRGetRequests	udr_rest_request_total{Method="GET"}	Total number of GET requests received by nudr-drservice (UDR related)
		UDRGetSuccessResponses	udr_rest_response_success_total{Method="GET"}	Total number of GET success responses sent by nudr-drservice (UDR related)
		UDRGetFailureResponses	udr_rest_response_failure_total{Method="GET"}	Total number of GET failure responses sent by nudr-drservice (UDR related)
	DELETE	UDRDeleteRequests	udr_rest_request_total{Method="DELETE"}	Total number of DELETE requests received by nudr-drservice (UDR related)
		UDRDeleteSuccessResponses	udr_rest_response_success_total{Method="DELETE"}	Total number of DELETE success responses sent by nudr-drservice (UDR related)
		UDRDeleteFailureResponses	udr_rest_response_failure_total{Method="DELETE"}	Total number of DELETE failure responses sent by nudr-drservice (UDR related)
	POST	UDRPostRequests	udr_rest_request_total{Method="POST"}	Total number of GET requests received by nudr-drservice (UDR related)
		UDRPostSuccessResponses	udr_rest_response_success_total{Method="POST"}	Total number of GET success responses sent by nudr-drservice (UDR related)
		UDRPostFailureResponses	udr_rest_response_failure_total{Method="POST"}	Total number of GET failure responses sent by nudr-drservice (UDR related)
Schema Validation	-	UDRSchemaValidationSuccess	udr_schema_validation_success_total	Total success count of schema validations. Done for all operations with body.

Category	Sub-category	Description	Metric Name	Notes
		UDRSchemaValidationFailure	udr_schema_validation_failure_total	Total failure count of schema validations. Done for all operations with body.
Schema Versioning	PUT	UDRPutSchemaSuccess	udr_schema_operations_success_total{Method="PUT"}	Total number of successful schema updates processed by nudr-drservice
		UDRPutSchemaFailure	udr_schema_operations_failure_total{Method="PUT"}	Total number of failure schema updates processed by nudr-drservice
	GET	UDRGetSchemaSuccess	udr_schema_operations_success_total{Method="GET"}	Total number of successful schema GET operations processed by nudr-drservice
		UDRGetSchemaFailure	udr_schema_operations_failure_total{Method="GET"}	Total number of failure schema GET operations processed by nudr-drservice
	DELETE	UDRDeleteSchemaSuccess	udr_schema_operations_success_total{Method="DELETE"}	Total number of successful schema deletion processed by nudr-drservice
		UDRDeleteSchemaFailure	udr_schema_operations_failure_total{Method="DELETE"}	Total number of successful schema deletion processed by nudr-drservice
DB	Reads	DbReadSuccess	udr_db_operations_success_total{Method="READ"}	Total number of successful DB reads performed by nudr-drservice on the back end NDB Cluster
		DbReadFailure	udr_db_operations_failure_total{Method="READ"}	Total number of failed DB reads performed by nudr-drservice on the back end NDB Cluster
	Writes/Updates	DbUpdateSuccess	udr_db_operations_success_total{Method="UPDATE"}	Total number of successful DB writes performed by nudr-drservice on the back end NDB Cluster

Category	Sub-category	Description	Metric Name	Notes
		DbUpdateFailure	udr_db_operations_failure_total{Method="UPDATE"}	Total number of failed DB writes performed by nudr-drservice on the back end NDB Cluster
	Transaction	DbTotalTxnAttempted	udr_total_transactions_total	Total number of successful DB transactions (includes all DB operations) performed by nudr-drservice on the back end NDB Cluster
		DbTxnSuccess	udr_successfull_transactions_total	Total number of failed DB transactions (includes all DB operations) performed by nudr-drservice on the back end NDB Cluster
Rest Controller UDSF	Aggregate	UDSFTotalRequests	udsf_rest_allrequests_total	Total number of requests received by nudr-drservice (All UDSF operations)
		UDSFTotalResponse	udsf_rest_allresponse_total	Total number of responses sent by nudr-drservice (All UDSF operations)
		UDSFTotalSuccessResponses	udsf_rest_successResponse_total	Total number of success responses sent by nudr-drservice (All UDSF operations)
		UDSFTotalFailureResponses	udsf_rest_failureResponse_total	Total number of failure responses sent by nudr-drservice (All UDSF operations)
	PUT	UDSFPutRequests	udsf_rest_request_total{Method="PUT"}	Total number of PUT requests received by nudr-drservice (UDSF related)

Category	Sub-category	Description	Metric Name	Notes
		UDSFPutSuccessResponses	udsf_rest_response_success_total{Method="PUT"}	Total number of PUT success responses sent by nudr-drservice (UDSF related)
		UDSFPutFailureResponses	udsf_rest_response_failure_total{Method="PUT"}	Total number of PUT failure responses sent by nudr-drservice (UDSF related)
	GET	UDSFGetRequests	udsf_rest_request_total{Method="GET"}	Total number of GET requests received by nudr-drservice (UDSF related)
		UDSFGETSuccessResponses	udsf_rest_response_success_total{Method="GET"}	Total number of GET success responses sent by nudr-drservice (UDSF related)
		UDSFGETFailureResponses	udsf_rest_response_failure_total{Method="GET"}	Total number of GET failure responses sent by nudr-drservice (UDSF related)
	DELETE	UDSFDeleteRequests	udsf_rest_request_total{Method="DELETE"}	Total number of DELETE requests received by nudr-drservice (UDSF related)
		UDSFDeleteSuccessResponses	udsf_rest_response_success_total{Method="DELETE"}	Total number of DELETE success responses sent by nudr-drservice (UDSF related)
		UDSFDeleteFailureResponses	udsf_rest_response_failure_total{Method="DELETE"}	Total number of DELETE failure responses sent by nudr-drservice (UDSF related)
Subscription	Aggregate	UDRSubscriptionRequests	nudr_dr_all_subscription_requests_total	Total number of subscription requests received by nudr-drservice (All operations)
		UDRSubscriptionRequestSuccessful	nudr_dr_all_subscription_responses_total	Total number of successful subscription request processed successfully by nudr-drservice (All operations)

Category	Sub-category	Description	Metric Name	Notes
		UDRSubscriptionResponseSuccessful	nudr_dr_all_subscription_successful_responses_total	Total number of successful subscription response sent by nudr-drservice (All operations)
		UDRSubscriptionResponseFailure	nudr_dr_all_subscription_failed_responses_total	Total number of successful subscription response sent by nudr-drservice (All operations)
	POST	UDRPostSubscriptionRequests	nudr_dr_subscription_request_total{Method="POST"}	Total number of POST Subscription Requests received by nudr-drservice
		UDRPostSubscriptionResponseSuccess	nudr_dr_subscription_successful_response_total{Method="POST"}	Total number of successful POST Subscription response sent by nudr-drservice
		UDRPostSubscriptionResponseFailure	nudr_dr_subscription_failed_response_total{Method="POST"}	Total number of failed POST Subscription response sent by nudr-drservice
	PUT	UDRPutSubscriptionRequests	nudr_dr_subscription_request_total{Method="PUT"}	Total number of PUT Subscription Requests received by nudr-drservice
		UDRPutSubscriptionResponseSuccess	nudr_dr_subscription_successful_response_total{Method="PUT"}	Total number of successful PUT Subscription response sent by nudr-drservice
		UDRPutSubscriptionResponseFailure	nudr_dr_subscription_failed_response_total{Method="PUT"}	Total number of failed PUT Subscription response sent by nudr-drservice
	GET	UDRGetSubscriptionRequests	nudr_dr_subscription_request_total{Method="GET"}	Total number of GET Subscription Requests received by nudr-drservice
		UDRGetSubscriptionResponseSuccess	nudr_dr_subscription_successful_response_total{Method="GET"}	Total number of successful GET Subscription response sent by nudr-drservice

Category	Sub-category	Description	Metric Name	Notes
		UDRGetSubscriptionResponseFailure	nudr_dr_subscription_failed_response_total{Method="GET"}	Total number of failed GET Subscription response sent by nudr-drservice
	DELETE	UDRDeleteSubscriptionRequests	nudr_dr_subscription_request_total{Method=”DELETE"}	Total number of DELETE Subscription Requests received by nudr-drservice
		UDRDeleteSubscriptionResponseSuccess	nudr_dr_subscription_successful_response_total{Method="DELETE"}	Total number of successful DELETE Subscription response sent by nudr-drservice
		UDRDeleteSubscriptionResponseFailure	nudr_dr_subscription_failed_response_total{Method="DELETE"}	Total number of failed DELETE Subscription response sent by nudr-drservice
Notification	Notifications to notification service	UDRNotificationRequests	nudr_dr_notif_signal_sent_total	Total notification sent successfully by nudr-drservice to nudr-notify-service
		UDRNotificationFailure	nudr_dr_notif_signal_failed_notifications_total	Total notification failed to send to nudr-notify-service
		UDRUpdateNotificationSuccess	nudr_dr_notif_signal_notification_update_success_total	Total update notifications sent successfully to nudr-notify-service
		UDRUpdateNotificationFailure	nudr_dr_notif_signal_notification_update_failed_total	Total update notifications failed to send to nudr-notify-service
		UDRDeleteNotificationSuccess	nudr_dr_notif_signal_notification_delete_success_total	Total delete notifications sent successfully to nudr-notify-service
		UDRDeleteNotificationFailure	nudr_dr_notif_signal_notification_delete_failed_total	Total delete notifications failed to send to nudr-notify-service

Category	Sub-category	Description	Metric Name	Notes
NF TYPE METRICS		UDR NF Type Specific Metrics	udr_nftype_specific_requests{NFType="PCF/UDM/SLF"}	Total number of requests that udr recevies from different nf types like ex: PCF,UDM,SLF
		UDR NF Type Specific Metrics - Method wise	udr_nftype_specific_requests{NFType="PCF/UDM/SLF",Method="GET/POST/PUT/DELETE/PATCH"}	Total number of requests that udr receives from different nf types along with method types
		UDR NF Type Specific Metrics - Provisioning Requests	udr_nftype_specific_requests{NFType="PCF/UDM/SLF",ProvRequest="Y/N"}	Total number of provisioning requests that will recevie udr specific to nf type
		UDR NF Type Specific Metrics - Method wise and Provisioning Requests	udr_nftype_specific_requests{NFType="PCF/UDM/SLF",Method="GET/POST/PUT/DELETE/PATCH",ProvRequest="Y/N"}	Total number of requests specifc to each nf type,method type and provisiong requests
		UDR NF Type Specific Metrics - Success Response	udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"2.*"}	Total number of success responses for each nf type that udr sent
		UDR NF Type Specific Metrics - Failure Response	udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"4.*"} +udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"5.*"}	Total number of failure responses for each nf type that udr sent
		UDR NF Type Specific Metrics - Success Response for provisioning requests	udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"2.*",ProvRequest="Y"}	Total number of success responses of provisioing requests for each nf type

Category	Sub-category	Description	Metric Name	Notes
		UDR NF Type Specific Metrics - Failure Response for provisioning requests	udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"4.*",ProvRequest="Y"} +udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"5.*",ProvRequest="Y"}	Total number of failure responses of provisioing requests for each nf type
		UDR NF Type Specific Metrics - Success response method wise	udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"2.*",Method="GET/POST/PUT/PATCH/DELETE"}	Total number of success responses of each method type of nf type
		UDR NF Type Specific Metrics - Failure response method wise	udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"4.*",Method="GET/POST/PUT/PATCH/DELETE"} +udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"5.*",Method="GET/POST/PUT/PATCH/DELETE"}	Total number of failure responses of each method type of nf type
NF TYPE - SUBSCRIPTION METRICS		UDR NF Type Subscription Metrics	nudr_dr_nf_type_subscription_request{NFType="PCF/UDM"}	Total number of subscription requests for each nf type i.e PCF,UDM
		UDR NF Type Subscription Metrics - Method Wise	nudr_dr_nf_type_subscription_request{NFType="PCF/UDM",Method="POST/PUT/DELETE/GET"}	Total number of subscription requests for each nf type as well as each method type

Category	Sub-category	Description	Metric Name	Notes
		UDR NF Type Subscription Metrics - Success Response	nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode="2xx"}	Total number of success responses of subscriptions
		UDR NF Type Subscription Metrics - Failure Response	nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*"}+ nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*"}	Total number of failure responses of subscriptions
		UDR NF Type Subscription Metrics - Success Response method wise	nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode="2xx",Method="POST/PUT/DELETE/GET/PATCH"}	Total number of subscription success based on nf type,method type.
		UDR NF Type Subscription Metrics - Failure Response Method wise	nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*",Method="POST/PUT/DELETE/GET/PATCH"} +nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*",Method="POST/PUT/DELETE/GET/PATCH"}	Total number of subscriptios failed based on nf type,method type.

Following metrics are applicable to **nudr-notify-service**.

Category	Sub-category	Description	Metric Name	Notes
Notification	Notifications received on notify service	Notifyservice_UDRNotificationRequests	nudr_notif_notifications_signals_received_total	Total number of notifications received on nudr-notify-service from nudr-drservice

Category	Sub-category	Description	Metric Name	Notes
		Notifyservice_UDRFailedNotificationsProcessing	nudr_notif_notifications_processing_failed_total	Total number of notification failure processing on nudr-notify-service
	Notifications to NF	Notifyservice_UDRNotificationsSent	nudr_notif_notifications_sent_total	Total number of notifications sent to network
		Notifyservice_UDRSendNotificationFailures	nudr_notif_notifications_send_fail_total	Total number of notifications failed to send to network
		Notifyservice_UDRNotificationsMarkedRetry	nudr_notif_notifications_marked_for_retry_total	Total number of notifications marked for retry after notification failure response
		NotifyService_UDRNotificationResponses	nudr_notif_notifications_ack_total	Total number of responses for notifications sent to network
		NotifyService_UDRNotificationSuccessResponses	nudr_notif_notifications_ack_2xx_total	Total number of success responses for notifications sent to network
		NotifyService_UDRNotification4xxResponses	nudr_notif_notifications_ack_4xx_total	Total number of 4xx error responses for notifications sent to network
		NotifyService_UDRNotification5xxResponses	nudr_notif_notifications_ack_5xx_total	Total number of 5xx error responses for notifications sent to network
NF TYPE - NOTIFICATION METRICS		UDR NF Type Notification Metrics - Notifcations received to notify service from different nf type	nudr_notif_notifications_nftype_signals_received{NFType="UDM/PCF"}	Total number of notifications received to sent out to udm or pcf
		UDR NF Type Notification Metrics - Notifcations received to notify service from nf types	nudr_notif_notifications_nftype_signals_received{NFType="UDM/PCF",Method="UPDATE/DELETE"}	Total number of notifications received to sent out to udm or pcf with update/delete requests.

Following metrics are applicable to **nudr-nrf-client** service.

Category	Sub-category	Description	Metric Name	Notes
	Deregistration	UDRNRFDeRegistrationRequestsSent	udr_nrf_deregistration_requests_total	Total number of registration requests sent by nrf-client-service to NRF
		UDRNRFDeRegistrationRequestsSuccessfull	udr_nrf_deregistration_success_total	Total number for successfully processed deregistration requests
Heartbeat towards NRF	Heartbeat	UDRNRFHeartBeatUpdateRequestsSent	udr_nrf_heartBeatUpdate_requests_total	Total number of heartbeat requests sent by nrf-client-service toward NRF to keep the status of UDR active
		UDRNRFHeartBeatUpdateRequestsSuccessfull	udr_nrf_heartBeatUpdate_success_total	Total number of successfully processed heartbeat messages
	LivenessProbe	UDRNRFLivenessProbeFailure	udr_nrf_livenessProbe_failure_total	Total number of failure attempts of liveness probe check on the udr micro services before registration
UDR Registration with NRF	Registration	UDRNRFRegistrationRequestsSent	udr_nrf_registration_requests_total	Total number of registration requests sent by nrf-client-service to NRF
		UDRNRFRegistrationRequestsSuccessfull	udr_nrf_registration_success_total	Total number for successfully processed registration requests

Following table provides information about the **ocudr-ingress-gateway** micro service for OCUDR.

KPI Details	Service Operation	KPI	Response Code	Notes
No of Requests/sec	All	UDR Ingress Request Rate	Not Applicable	sum(irate(oc_ingressgateway_http_requests_total[5m]))
No of Responses/sec	All	UDR Ingress Response Rate	Not Applicable	sum(irate(oc_ingressgateway_http_responses_total[5m]))

KPI Details	Service Operation	KPI	Response Code	Notes
No of Successful Responses of each type/No of Successful Responses per second	PostRequest	rate of Post Requests with success response	201	sum(irate(oc_ingressgateway_http_responses_total{Method="POST",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m]))
	PutRequests	rate of Put Requests with success response	201	sum(irate(oc_ingressgateway_http_responses_total{Method="PUT",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m]))
	GetRequests	rate of Get Requests with success response	200	sum(irate(oc_ingressgateway_http_responses_total{Method="GET",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m]))
	DeleteRequests	rate of Delete Requests with success response	204	sum(irate(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m]))
	PatchRequests	rate of Patch Requests with success response	204	sum(irate(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m]))

KPI Details	Service Operation	KPI	Response Code	Notes
No of 4xx responses of each type/No of 4xx responses per second	PostRequest	rate of PostRequests failed with 4xx	4xx	sum(irate(oc_ingressgateway_http_responses_total{Method="POST",Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m]))
	PutRequests	rate of PutRequests failed with 4xx	4xx	sum(irate(oc_ingressgateway_http_responses_total{Method="PUT",Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m]))
	GetRequests	rate of GetRequests failed with 4xx	4xx	sum(irate(oc_ingressgateway_http_responses_total{Method="GET",Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m]))
	DeleteRequests	rate of DeleteRequests failed with 4xx	4xx	sum(irate(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m]))
	PatchRequests	rate of PatchRequests failed with 4xx	4xx	sum(irate(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m]))

KPI Details	Service Operation	KPI	Response Code	Notes
No of 5xx responses of each type/No of 5xx responses per second	PostRequest	rate of PostRequests failed with 5xx	5xx	(sum(irate(oc_ingressgateway_http_requests_total{Method="POST"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="POST"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m])))
	PutRequests	rate of PutRequests failed with 5xx	5xx	(sum(irate(oc_ingressgateway_http_requests_total{Method="PUT"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="PUT"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m])))
	GetRequests	rate of GetRequests failed with 5xx	5xx	(sum(irate(oc_ingressgateway_http_requests_total{Method="GET"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="GET"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m])))

KPI Details	Service Operation	KPI	Response Code	Notes
	DeleteRequests	rate of DeleteRequests failed with 5xx	5xx	(sum(irate(oc_ingressgateway_http_requests_total{Method="DELETE"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="DELETE"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m])))
	PatchRequests	rate of PatchRequests failed with 5xx	5xx	(sum(irate(oc_ingressgateway_http_requests_total{Method="PATCH"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="PATCH"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m])))
Percentage of success requests of each type of operations	PostRequest	Percentage of Post Requests with success response	201	sum(oc_ingressgateway_http_responses_total{Method="POST",Status=~"2.*"})/sum(oc_ingressgateway_http_responses_total)
	PutRequests	Percentage of Put Requests with success response	2xx	sum(oc_ingressgateway_http_responses_total{Method="PUT",Status=~"2.*"})/sum(oc_ingressgateway_http_responses_total)

KPI Details	Service Operation	KPI	Response Code	Notes
	GetRequests	Percentage of Get Requests with success response	200	sum(oc_ingressgateway_http_responses_total{Method="GET",Status=~"2.*"})/ sum(oc_ingressgateway_http_responses_total)
	DeleteRequests	Percentage of Delete Requests with success response	204	sum(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"2.*"})/ sum(oc_ingressgateway_http_responses_total)
	PatchRequests	Percentage of Patch Requests with success response	204	sum(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"2.*"})/ sum(oc_ingressgateway_http_responses_total)
Percentage of 4xx requests of each type of operations	PostRequest	Percentage of PostRequests failed with 4xx	4xx	sum(oc_ingressgateway_http_responses_total{Method="POST",Status=~"4.*"})/ sum(oc_ingressgateway_http_responses_total{Status=~"4.*"})
	PutRequests	Percentage of PutRequests failed with 4xx	4xx	sum(oc_ingressgateway_http_responses_total{Method="PUT",Status=~"4.*"})/ sum(oc_ingressgateway_http_responses_total{Status=~"4.*"})
	GetRequests	Percentage of GetRequests failed with 4xx	4xx	sum(oc_ingressgateway_http_responses_total{Method="GET",Status=~"4.*"})/ sum(oc_ingressgateway_http_responses_total{Status=~"4.*"})

KPI Details	Service Operation	KPI	Response Code	Notes
	DeleteRequests	Percentage of DeleteRequests failed with 4xx	4xx	sum(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"4.*"})/ sum(oc_ingressgateway_http_responses_total{Status=~"4.*"})
	PatchRequests	Percentage of PatchRequests failed with 4xx	4xx	sum(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"4.*"})/ sum(oc_ingressgateway_http_responses_total{Status=~"4.*"})
Percentage of 5xx requests of each type of operations	PostRequest	Percentage of PostRequests failed with 5xx	5xx	(sum(oc_ingressgateway_http_requests_total{Method="POST"})- sum(oc_ingressgateway_http_responses_total{Method="POST"}))/ (sum(oc_ingressgateway_http_requests_total)- sum(oc_ingressgateway_http_responses_total))
	PutRequests	Percentage of PutRequests failed with 5xx	5xx	(sum(oc_ingressgateway_http_requests_total{Method="PUT"})- sum(oc_ingressgateway_http_responses_total{Method="PUT"}))/ (sum(oc_ingressgateway_http_requests_total)- sum(oc_ingressgateway_http_responses_total))

KPI Details	Service Operation	KPI	Response Code	Notes
	GetRequests	Percentage of GetRequests failed with 5xx	5xx	(sum(oc_ingressgateway_http_requests_total{Method="GET"})-sum(oc_ingressgateway_http_responses_total{Method="GET"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total))
	DeleteRequests	Percentage of DeleteRequests failed with 5xx	5xx	(sum(oc_ingressgateway_http_requests_total{Method="DELETE"})-sum(oc_ingressgateway_http_responses_total{Method="DELETE"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total))
	PatchRequests	Percentage of PatchRequests failed with 5xx	5xx	(sum(oc_ingressgateway_http_requests_total{Method="PATCH"})-sum(oc_ingressgateway_http_responses_total{Method="PATCH"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total))

A
Rollback Instructions for PCF Data

In this Appendix, you will learn to revert to 29.519 v15.3.0 for PCF Data.

> **Note:**
>
> In a deployed UDR, you must configure only 1 version of PCF data (either v16.2.0 or v15.3.0). By default, UDR 1.6.0 supports 29.519 v16.2.0 version of PCF data. If there are multiple versions, it will create inconsistency in the json data stored on UDR.

> **Note:**
>
> If the user wishes to be compatible with 29.519 v15.3 of PCF data, it must follow instructions while deploying UDR and should not change, if there are some subscribers provisioned using any older schema.

Schema Rollback to 29.519 15.3.0
A script (***rollbackPCFschema_15_3.py***) is provided for this procedure (as part of the customer documentation). The steps to execute the script are:

1. Open the script and edit the following details as per the udrdb configurations.

   ```
   mydb = mysql.connector.connect(
   host="localhost",
   user="root",
   passwd="xxxxxxxx",
   database="udrdb")
   ```

2. Execute the below command:

   ```
   python rollbackPCFschema_15_3.py
   ```