**Impacts in SDN-R for OOF SON PCI optimization use case**

# **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Change(s)** |
| V\_0\_7 | Sept 20, 2018 | Sandeep Shah & Devendra Chauhan | Updated YANG model and pertinent nodes to be used in the POC |
| V\_0\_5 | Sep 18th 2018 | Swaminathan S & Saravanan A (Wipro) | Added error code details in SDNR ResponseUpdated message contents as per latest yang model. A few other changes done, and highlighted using ‘Track changes’. |
| v\_0\_4 | Sep 12th 2018 | Saravanan A (Wipro) | Updated data exchanged between SDNR and PCI MS(Section 2.1.1) , Policy(Section 2.1.2 and 2.1.3) |
| v\_0\_3 | Aug 31st 2018 | Swaminathan S & Saravanan A (Wipro) | Updates done to Interfaces and DMaaP message contents based on comments received in SDN-R call on 29th. |
| v\_0\_2 | Aug 29th 2018 | Swaminathan S & Saravanan A (Wipro) | Updates done based on internal review. |
| v\_0\_1 | Aug 29th 2018 | Swaminathan S & Saravanan A (Wipro) | First version |

# **Message Flow**





# **Interface impacts**



## **DMaaP messages**

### **SDN-R to PCI-HMS (Step 4a)**

**DMaaP Topic Name: PCI-NOTIF-TOPIC-NGHBR-LIST-CHANGE-INFO**

The *Payload* part can be made to align with the Yang model of the notification from RAN to SDN-R, provided the pnfName is also present. Otherwise the pnfName has to be included by SDN-R.

**Assumption**: The RAN could send neighbor list change for more than 1 cell in a single notification to PCI-MS (as the RAN netconf server could have more than 1 cell associated with it).

{

  "requestID": "9d2d790e-a5f0-11e8-98d0-529269fb1459",

  "AAI": {},

  "from": "SDNR",

  "version": "1.0.2",

  "Action": "NeighborListModified",

  "Payload": {

    "RadioAccess": [

      {

          "FAPService": {

            "alias": "Cell1",

 "X0005b9Lte" : {

            "phyCellIdInUse" : 35,

            "pnfName" : "CU-1"

 },

            "CellConfig": {

              "LTE": {

                "RAN": {

                  "Common": {

                    "CellIdentity": "1"

                  },

                  "RF": {

                    "PhyCellID": 35

                  },

                  "NeighborList": [

                    {

 "pnfName" : "CU-2",

                      "enable": true,

                      "alias": "Cell2",

                      "mustInclude": true,

                      "plmnid": "123456",

                      "cid": "2",

                      "phyCellId": 22,

                      "blacklisted": false

                    }

                  ]

                }

              }

            }

          }      },

      {

          "FAPService": {

            "alias": "Cell2",

 "X0005b9Lte" : {

            "phyCellIdInUse" : 22,

            "pnfName" : "CU-2"

 },

            "CellConfig": {

              "LTE": {

                "RAN": {

                  "Common": {

                    "CellIdentity": "2"

                  },

                  "RF": {

                    "PhyCellID": 22

                  },

                  "NeighborList": [

                    {

 "pnfName" : "CU-1",

                      "enable": true,

                      "alias": "Cell1",

                      "mustInclude": true,

                      "plmnid": "123456",

                      "cid": 1,

                      "phyCellId": 35,

                      "blacklisted": false

                    }

                  ]

                }

              }

            }

          }

      }

    ]

  }

}

### **Policy to SDN-R (Step 10)**

**DMaaP Topic Name: SDNR-CL**

This message is sent from Policy to SDN-R in Step 10 to trigger the PCI configuration changes. The payload will be copied as is from the message received from PCI-HMS in Step 9, which, in turn, will be aligned to the yang model definition

{

 "body": {

 "input": {

 "CommonHeader": {

 "TimeStamp": "2018-09-12T12:11:49.220Z",

 "APIVer": "1.0",

 "RequestID": "664be3d2-6c12-4f4b-a3e7-c349acced200",

 "SubRequestID": "1",

 "RequestTrack": {},

 "Flags": {}

 },

 "Action": "ModifyConfig",

 "Payload": "{ \"radioAccess\": [ { \"fapService\": { \"alias\": \"Network1\", \"x0005b9Lte\": { \"phyCellIdInUse\": 35, \"pnfName\": \"CU-1\" }, \"cellConfig\": { \"lte\": { \"ran\": { \"common\": { \"cellIdentity\": \"1\" }, \"rf\": { \"phyCellID\": \"35\" } } } } } }, { \"fapService\": { \"alias\": \"Network1\", \"x0005b9Lte\": { \"phyCellIdInUse\": 22, \"pnfName\": \"CU-2\" }, \"cellConfig\": { \"lte\": { \"ran\": { \"common\": { \"cellIdentity\": \"2\" }, \"rf\": { \"phyCellID\": \"22\" } } } } } } ] }"

 }

 },

 "version": "1.0",

 "rpc-name": "restart",

 "correlation-id": "664be3d2-6c12-4f4b-a3e7-c349acced200-1",

 "type": "request"

}

### **SDN-R to Policy**

**DMaaP Topic Name: SDNR-CL-RSP**

This DMaaP message is sent by SDN-R to Policy as acknowledgement of Step 10, after execution of the actions recommended by Policy in Step 10.

There is an overall status code, as well as a status code per PNF.

{

 "body": {

 "output": {

 "CommonHeader": {

 "TimeStamp": "2018-09-12T12:11:49.220Z",

 "APIVer": "1.0",

 "RequestID": "664be3d2-6c12-4f4b-a3e7-c349acced200",

 "SubRequestID": "1",

 "RequestTrack": {},

 "Flags": {}

 },

 "Status": {

 "Code": 400,

 "Value": "Modify is partially successful"

 },

 "Payload": "{ \"radioAccess\": [ { \"fapService\": { \"alias\": \"Network1\",\"x0005b9Lte\": { \"phyCellIdInUse\": 35, \"pnfName\": \"CU-1\" },\"cellConfig\": { \"lte\": { \"ran\": { \"common\": { \"cellIdentity\": \"1\" }, \"rf\": { \"phyCellID\": \"35\" } } } }, \"Status\": { \"Code\": 200, \"Value\": \"SUCCESS\" } } }, { \"fapService\": { \"alias\": \"Network1\", \"x0005b9Lte\": { \"phyCellIdInUse\": 22, \"pnfName\": \"CU-2\" }, \"cellConfig\": { \"lte\": { \"ran\": { \"common\": { \"cellIdentity\": \"2\" }, \"rf\": { \"phyCellID\": \"22\" } } } }, \"Status\": { \"Code\": 200, \"Value\": \"SUCCESS\" } } } ] }"

 }

 },

 "version": "1.0",

 "rpc-name": "ModifyConfig",

 "correlation-id": "664be3d2-6c12-4f4b-a3e7-c349acced200-1",

 "type": "request"

}

The Status Codes are defined as below:

|  |  |  |
| --- | --- | --- |
| Status Code(int) | Status Category | Description |
| 100 | Accepted | May not be used in current use case |
| 200 | Success | When all requests are successful |
| 300 to 313 | Reject | When request format is invalid |
| 400 | Error | When all requests are failed at SDNR level |
| 401 to 406, 450 | Failure | When all requests are failed at RAN network level |
| 500 | Partial Success | When some requests are successful |
| 501 to 599 | Partial Failure | When some requests are failed |

## **REST API calls**

The REST API calls below could be used by PCI-MS as well as by OOF for fetching the relevant neighbor list and PCI information from SDN-R config database.

### **/** **SDNCConfigDBAPI /getCellList**

 "/SDNCConfigDBAPI/getCellList": {

 "get": {

 "tags": [

 "ran-db-config-controller"

 ],

 "summary": "getCellList",

 "operationId": "getCellListUsingGET",

 "consumes": [

 "application/json"

 ],

 "produces": [

 "application/json"

 ],

 "parameters": [

 {

 "name": "networkId",

 "in": "query",

 "description": "networkId",

 "required": true,

 "type": "string"

 },

 {

 "name": "ts",

 "in": "query",

 "description": "ts",

 "required": false,

 "type": "string",

 "format": "date-time"

 }

 ],

 "responses": {

 "200": {

 "description": "OK",

 "schema": {

 "type": "array",

 "items": {

 "type": "string"

 }

 }

 },

 "401": {

 "description": "Unauthorized"

 },

 "403": {

 "description": "Forbidden"

 },

 "404": {

 "description": "Not Found"

 }

 }

 }

 }

### **/SDNCConfigDBAPI/getNbrList**

 "/SDNCConfigDBAPI/getNbrList": {

 "get": {

 "tags": [

 "ran-db-config-controller"

 ],

 "summary": "getNbrList",

 "operationId": "getNbrListUsingGET",

 "consumes": [

 "application/json"

 ],

 "produces": [

 "application/json"

 ],

 "parameters": [

 {

 "name": "cellId",

 "in": "query",

 "description": "cellId",

 "required": true,

 "type": "string"

 },

 {

 "name": "ts",

 "in": "query",

 "description": "ts",

 "required": true,

 "type": "string",

 "format": "date-time"

 }

 ],

 "responses": {

 "200": {

 "description": "OK",

 "schema": {

 "type": "array",

 "items": {

 "$ref": "#/definitions/NbrList"

 }

 }

 },

 "401": {

 "description": "Unauthorized"

 },

 "403": {

 "description": "Forbidden"

 },

 "404": {

 "description": "Not Found"

 }

 }

 }

 },

### **/SDNCConfigDBAPI/getPCI**

 "/SDNCConfigDBAPI/getPCI": {

 "get": {

 "tags": [

 "ran-db-config-controller"

 ],

 "summary": "getPCI",

 "operationId": "getPCIUsingGET",

 "consumes": [

 "application/json"

 ],

 "produces": [

 "application/json"

 ],

 "parameters": [

 {

 "name": "cellId",

 "in": "query",

 "description": "cellId",

 "required": true,

 "type": "string"

 },

 {

 "name": "ts",

 "in": "query",

 "description": "ts",

 "required": true,

 "type": "string",

 "format": "date-time"

 }

 ],

 "responses": {

 "200": {

 "description": "OK",

 "schema": {

 "type": "integer",

 "format": "int64"

 }

 },

 "401": {

 "description": "Unauthorized"

 },

 "403": {

 "description": "Forbidden"

 },

 "404": {

 "description": "Not Found"

 }

 }

 }

 },

### **/SDNCConfigDBAPI/getPnfName**

 "/SDNCConfigDBAPI/getPnfName": {

 "get": {

 "tags": [

 "ran-db-config-controller"

 ],

 "summary": "getPnfName",

 "operationId": "getPnfNameUsingGET",

 "consumes": [

 "application/json"

 ],

 "produces": [

 "application/json"

 ],

 "parameters": [

 {

 "name": "cellId",

 "in": "query",

 "description": "cellId",

 "required": true,

 "type": "string"

 },

 {

 "name": "ts",

 "in": "query",

 "description": "ts",

 "required": true,

 "type": "string",

 "format": "date-time"

 }

 ],

 "responses": {

 "200": {

 "description": "OK",

 "schema": {

 "type": "string"

 }

 },

 "401": {

 "description": "Unauthorized"

 },

 "403": {

 "description": "Forbidden"

 },

 "404": {

 "description": "Not Found"

 }

 }

 }

 }

 },

## **Yang models (snippets)**

The relevant Yang model tree snippet for this PCI Optimization PoC is shown below, and pertinent YANG nodes that will be used and populated are highlighted in YELLOW:

module: bbf-tr-196-2-0-3-full

 +--rw radio-access

 +--rw fap-service\* [alias]

 +--rw alias string

 +--rw device-type? enumeration

 +--ro dn-prefix? string

 +--rw x-0005b9-lte

 | +--rw phy-cell-id-in-use? uint64

 | +--rw pnf-name? string

...

 +--rw cell-config

...

 | +--rw lte

 | | +--rw tunnel-number-of-entries? uint64

 | | +--rw lte-tunnel\* [tunnel-ref]

 | | | +--ro enable? boolean

 | | | +--rw alias? string

 | | | +--rw tunnel-ref string

 | | | +--ro plmnid\* string

...

 | | +--rw lte-ran

 | | +--rw lte-ran-common

 | | | +--ro cell-identity? string

...

 | | +--rw lte-ran-rf

 | | | +--ro earfcndl\* uint16

 | | | +--ro earfcnul\* uint16

 | | | +--ro freq-band-indicator? uint8

 | | | +--ro dl-bandwidth\* uint8

 | | | +--ro ul-bandwidth\* uint8

 | | | +--ro reference-signal-power\* string

 | | | +--ro phy-cell-id\* string

 | | | +--ro psch-power-offset\* string

...

 | | +--rw lte-ran-neighbor-list

 | | | +--rw max-lte-cell-entries? uint64

 | | | +--rw lte-cell-number-of-entries? uint64

 | | | +--rw lte-ran-neighbor-list-lte-cell\* [plmnid cid]

 | | | | +--ro enable? boolean

 | | | | +--rw alias? string

 | | | | +--ro must-include? boolean

 | | | | +--rw plmnid string

 | | | | +--rw cid string

 | | | | +--ro eutra-carrier-arfcn? uint16

 | | | | +--ro phy-cell-id? uint64

 | | | | +--ro q-offset? int64

 | | | | +--ro cio? int64

 | | | | +--ro rs-tx-power? int64

 | | | | +--ro blacklisted? boolean

...

 | | +--rw lte-ran-neighbor-list-in-use

 | | +--rw max-lte-cell-entries? uint64

 | | +--rw lte-cell-number-of-entries? uint64

 | | +--rw lte-ran-neighbor-list-in-use-lte-cell\* [plmnid cid]

 | | | +--rw plmnid string

 | | | +--rw cid string

 | | | +--rw eutra-carrier-arfcn? uint16

 | | | +--rw phy-cell-id? uint64

 | | | +--rw q-offset? int64

 | | | +--rw cio? int64

 | | | +--rw rs-tx-power? int64

 | | | +--rw blacklisted? boolean

...

Below table also includes paths (highlighted in grey) to pertinent YANG nodes:

module: bbf-tr-196-2-0-3-full (

 +--rw radio-access (/radio-access)

 +--rw fap-service\* [alias] (/radio-access/fap-service)

 +--rw alias string (/radio-access/fap-service/alias)

 +--rw x-0005b9-lte (/radio-access/fap-service/x-0005b9-lte)

 | +--rw phy-cell-id-in-use? uint64 (/radio-access/fap-service/x-0005b9-lte/phy-cell-id-in-use)

 | +--rw pnf-name? string (/radio-access/fap-service/x-0005b9-lte/pnf-name)

...

 +--rw cell-config (/radio-access/fap-service/cell-config)

...

 | +--rw lte (/radio-access/fap-service/cell-config/lte)

 | | +--rw tunnel-number-of-entries? uint64 (/radio-access/fap-service/cell-config/lte/tunnel-number-of-entries)

 | | +--rw lte-tunnel\* [tunnel-ref] (/radio-access/fap-service/cell-config/lte/lte-tunnel)

 | | | +--rw tunnel-ref string (/radio-access/fap-service/cell-config/lte/lte-tunnel/tunnel-ref)

 | | | +--ro plmnid\* string (/radio-access/fap-service/cell-config/lte/lte-tunnel/plmnid)

...

 | | +--rw lte-ran (/radio-access/fap-service/cell-config/lte/lte-ran)

 | | | +--ro cell-identity? String (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-common/cell-identity)

...

 | | +--rw lte-ran-rf (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-rf)

 | | | +--ro phy-cell-id\* string (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-rf/phy-cell-id)

...

 | | +--rw lte-ran-neighbor-list (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list)

 | | | +--rw lte-ran-neighbor-list-lte-cell\* [plmnid cid] (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list/lte-ran-neighbor-list-lte-cell)

 | | | | +--rw plmnid string (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list/lte-ran-neighbor-list-lte-cell/plmnid)

 | | | | +--rw cid string (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list/lte-ran-neighbor-list-lte-cell/cid)

 | | | | +--ro phy-cell-id? uint64 (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list/lte-ran-neighbor-list-lte-cell/phy-cell-id)

...

 | | +--rw lte-ran-neighbor-list-in-use (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list-in-use)

 | | +--rw lte-ran-neighbor-list-in-use-lte-cell\* [plmnid cid] (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list-in-use/lte-ran-neighbor-list-in-use-lte-cell)

 | | | +--rw plmnid string (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list-in-use/lte-ran-neighbor-list-in-use-lte-cell/plmnid)

 | | | +--rw cid string (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list-in-use/lte-ran-neighbor-list-in-use-lte-cell/cid)

 | | | +--rw phy-cell-id? uint64 (/radio-access/fap-service/cell-config/lte/lte-ran/lte-ran-neighbor-list-in-use/lte-ran-neighbor-list-in-use-lte-cell/phy-cell-id)

...

### **Neighbor-list change notification from RAN to SDN-R (Step 3)**

The corresponding Yang model tree format is shown below, along with pertinent nodes highlighted in YELLOW:

 +--rw radio-access

 +--rw fap-service\* [alias]

...

 | +--rw lte

...

 | | +--rw lte-ran

...

 | | +--rw lte-ran-rf

...

 | | +--rw lte-ran-neighbor-list

 | | | +--rw max-lte-cell-entries? uint64

 | | | +--rw lte-cell-number-of-entries? uint64

 | | | +--rw lte-ran-neighbor-list-lte-cell\* [plmnid cid]

 | | | | +--ro enable? boolean

 | | | | +--rw alias? string

 | | | | +--ro must-include? boolean

 | | | | +--rw plmnid string

 | | | | +--rw cid string

 | | | | +--ro eutra-carrier-arfcn? uint16

 | | | | +--ro phy-cell-id? uint64

 | | | | +--ro q-offset? int64

 | | | | +--ro cio? int64

 | | | | +--ro rs-tx-power? int64

 | | | | +--ro blacklisted? boolean

...

 | | +--rw lte-ran-neighbor-list-in-use

 | | +--rw max-lte-cell-entries? uint64

 | | +--rw lte-cell-number-of-entries? uint64

 | | +--rw lte-ran-neighbor-list-in-use-lte-cell\* [plmnid cid]

 | | | +--rw plmnid string

 | | | +--rw cid string

 | | | +--rw eutra-carrier-arfcn? uint16

 | | | +--rw phy-cell-id? uint64

 | | | +--rw q-offset? int64

 | | | +--rw cio? int64

 | | | +--rw rs-tx-power? int64

 | | | +--rw blacklisted? boolean

...

| +--ro blacklisted? boolean

### **Config change command from SDN-R to RAN to change PCI values (Step 11)**

The corresponding Yang model tree format is shown below, , along with pertinent nodes highlighted in YELLOW:

:

 +--rw radio-access

 +--rw fap-service\* [alias]

 +--rw alias string

 +--rw device-type? enumeration

 +--ro dn-prefix? string

 +--rw x-0005b9-lte

 | +--rw phy-cell-id-in-use? uint64

 | +--rw pnf-name? string

...

# **Code impacts**

## **RAN Simulator Initial Setup and Shutdown**

1. After each RAN Cell node is started, RAN Simulator Controller will send below mount request to SDNR to add the device into topology.

Ref: https://docs.opendaylight.org/en/stable-oxygen/user-guide/netconf-user-guide.html#spawning-new-netconf-connectors

curl -X PUT --trace-ascii FILE.txt -H 'Content-Type: application/xml' -H 'Accept: application/xml' -u admin:admin "http://<SDNR-IP>:8181/restconf/config/network-topology:network-topology/topology/topology-netconf/node/DU-1" -d '

<node

 xmlns="urn:TBD:params:xml:ns:yang:network-topology">

 <node-id>DU-1</node-id>

 <host

 xmlns="urn:opendaylight:netconf-node-topology">10.145.59.89

 </host>

 <port

 xmlns="urn:opendaylight:netconf-node-topology">50000

 </port>

 <username

 xmlns="urn:opendaylight:netconf-node-topology">admin

 </username>

 <password

 xmlns="urn:opendaylight:netconf-node-topology">admin

 </password>

 <tcp-only

 xmlns="urn:opendaylight:netconf-node-topology">false

 </tcp-only>

 <!-- non-mandatory fields with default values, you can safely remove these if you do not wish to override any of these values-->

 <reconnect-on-changed-schema

 xmlns="urn:opendaylight:netconf-node-topology">false

 </reconnect-on-changed-schema>

 <connection-timeout-millis

 xmlns="urn:opendaylight:netconf-node-topology">20000

 </connection-timeout-millis>

 <max-connection-attempts

 xmlns="urn:opendaylight:netconf-node-topology">0

 </max-connection-attempts>

 <between-attempts-timeout-millis

 xmlns="urn:opendaylight:netconf-node-topology">2000

 </between-attempts-timeout-millis>

 <sleep-factor

 xmlns="urn:opendaylight:netconf-node-topology">1.5

 </sleep-factor>

 <keepalive-delay

 xmlns="urn:opendaylight:netconf-node-topology">120

 </keepalive-delay>

</node> '

1. SDN-R setup Netconf session with RAN Node by exchanging hello messages.
2. Before stopping each RAN Cell node, RAN Simulator Controller will send below unmount request to SDNR to remove the device entry from topology.

Ref: https://docs.opendaylight.org/en/stable-oxygen/user-guide/netconf-user-guide.html#deleting-an-existing-connector

curl -X DELETE --trace-ascii FILE.txt -H 'Content-Type: application/xml' -H 'Accept: application/xml' -u admin:admin "http://<SDNR-IP>:8181/restconf/config/network-topology:network-topology/topology/topology-netconf/node/DU-1”

## **RAN sends neighbor-list change notification to SDN-R (Step 3)**

1. SDN-R receives a notification containing the neighbor list (list of cell-ids and the associated PCIs) from RAN. Note: A new Yang model to be defined for this notification, some implementation may be required in SDN-R to handle this new notification, as it is not a Netconf base notification (as defined in RFC 6470).

## **SDN-R sends neighbor-list change notification to PCI-HMS (Step 4a)**

1. SDN-R stores the notification contents in Config DB, prepares a DMaaP message as shown in Section 2.1.1 and publishes it to PCI-HMS in topic PCI-NOTIF-TOPIC-NGHBR-LIST-CHANGE-INFO.

## **Config DB API implementation (Steps 4b and 6)**

Implementation of the DB and the Rest API handling.

## **Policy sends config action control-loop message (Step 10)**

1. Upon reception of a control loop action request from Policy module over DMaaP in SDNR-CL topic, SDN-R extracts the details for each RAN node.
2. SDN-R then sends appropriate Netconf commands to the relevant RAN nodes (after checking their status, etc.).

## **RAN sends response for Config change action (response for Step 11)**

1. Upon reception of response (or timeout) from the RAN nodes, SDN-R prepares and sends a consolidated DMaaP response notification to Policy module in SDNR-CL-RSP topic.
2. It also updates its Config DB based on the response received from the RAN nodes.