Nokia FlowOne Catalog Driven Provisioning and Activation™ for Trident Optima-USPS Integration

Functional High-Level Design

|  |  |  |  |
| --- | --- | --- | --- |
|  | | |  |
| Author | De Torres, Rhoda (Nokia - PH/Taguig City) |
| Owner / Designated Access | Son, Edward (Nokia – PH/Taguig City) |
| Organization | Nokia |
| Approver | Amdocs |
|  |  |
|  |  |
| |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Change History | | | | | | | | | | | Version | Status | Date | Author | Owner | Reviewed by | Reviewed date | Approver | Approval date | Description of changes | | 1.0 | Draft | 09-12-2019 | De Torres, Rhoda | Son, Edward | Amdocs | DD-MM-YYYY | PLDT |  | Draft Version | | 1.1 | After Initial Review | 09-01-2020 | De Torres, Rhoda | Son, Edward | Amdocs | 09-01-2020 | PLDT |  | Updates from Amdocs Initial review | | 1.2 | After Final Review | 13-01-2020 | De Torres, Rhoda | Son, Edward | Amdocs | 15-01-2020 | PLDT |  | Updates from Amdocs Final review | | 1.3 | Initial Version | 22-01-2020 | De Torres, Rhoda | Son, Edward | Amdocs |  | PLDT |  | Initial Draft | | 1.4 | Phase1B | 25-02-2020 | De Torres, Rhoda | Son, Edward | Amdocs |  | PLDT |  | Requirements updates | | 1.5 | Phase1B | 10-06-2020 | De Torres, Rhoda | Son, Edward | Amdocs |  | PLDT |  | Final Version | | 1.6 | Intra MNP | 11-06-2020 | Reyes, Reybal | Son, Edward | Amdocs |  | PLDT |  | Added Intra MNP | | 1.7 | After review | 22-06-2020 | Reyes, Reybal | Son, Edward | Amdocs |  | PLDT |  | Applied changes base on the  review of Intra MNP | | 1.8 | CR | 02-09-2020 | De Torres, Rhoda | Son, Edward | Amdocs |  | PLDT |  | Added the changes listed in  Appendix D Change Tracker | | 1.9 | Inter MNP | 21-01-2021 | Lai, Kamhoong | Son, Edward | Amdocs |  | PLDT |  | Added Inter MNP. Updated section  3.1, 3.2, 3.4.3, 4.4.4, 4.6.8. | | 2.0 | InterMNP Non-DSA Blank SIM Support | 08-12-2021 | Lai, Kamhoong | Wongchamboon, Parin | Amdocs |  | PLDT |  | Added Inter MNP Non-DSA  related changes. Updated section  3.1, 4.4.5, 4.6.8 | |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  | | |  |
|  | | |  |

Contents

[1 About This Document 1](#_Toc89863627)

[1.1 Audience 2](#_Toc89863628)

[1.2 Terms and Concepts 2](#_Toc89863629)

[1.3 Typographic Conventions 3](#_Toc89863630)

[1.3.1 Abbreviations 3](#_Toc89863631)

[1.3.2 Terminology 4](#_Toc89863632)

[1.4 Related Documentation 4](#_Toc89863633)

[2 Executive Summary 5](#_Toc89863634)

[2.1 Success Factors 5](#_Toc89863635)

[3 Overview 6](#_Toc89863636)

[3.1 Services 6](#_Toc89863637)

[3.2 Network Context 8](#_Toc89863638)

[3.3 Processes 10](#_Toc89863639)

[3.3.1 User Interfaces 12](#_Toc89863640)

[3.3.2 Reporting 12](#_Toc89863641)

[3.3.3 Customized Reporting 12](#_Toc89863642)

[3.3.3.1 IRAFM Reports 13](#_Toc89863643)

[3.3.3.2 IRAFM Manifest File 13](#_Toc89863644)

[3.3.4 Localization 13](#_Toc89863645)

[3.3.5 FlowOne (InstantLink) Archive 13](#_Toc89863646)

[3.4 Integrations 14](#_Toc89863647)

[3.4.1 Northbound Interfaces 14](#_Toc89863648)

[3.4.1.1 Protocol 14](#_Toc89863649)

[3.4.1.2 Message Structure 14](#_Toc89863650)

[3.4.1.3 Batch API 17](#_Toc89863651)

[3.4.2 East/West Interfaces 17](#_Toc89863652)

[3.4.3 Southbound Interfaces 17](#_Toc89863653)

[3.4.4 Nokia Service Catalog 20](#_Toc89863654)

[4 Solution Architecture 21](#_Toc89863655)

[4.1 Product 21](#_Toc89863656)

[4.2 Target Solution Architecture 22](#_Toc89863657)

[4.3 Target Application Architecture 23](#_Toc89863658)

[4.4 High Level Process Workflow 24](#_Toc89863659)

[4.4.1 Phase 1a 25](#_Toc89863660)

[4.4.1.1 Activation of Wireless Subscriber 25](#_Toc89863661)

[4.4.1.2 Wireless Service Modification 26](#_Toc89863662)

[4.4.1.3 Suspend/Resume Wireless Subscriber 27](#_Toc89863663)

[4.4.1.4 Disconnect Wireless Subscriber 29](#_Toc89863664)

[4.4.2 Phase 1b 30](#_Toc89863665)

[4.4.2.1 Change Billing Cycle 30](#_Toc89863666)

[4.4.2.2 Change Ownership 31](#_Toc89863667)

[4.4.2.3 Bar/Unbar Wireless Subscriber 32](#_Toc89863668)

[4.4.2.4 IPTV 33](#_Toc89863669)

[4.4.2.5 SAT TV 34](#_Toc89863670)

[4.4.2.6 Notice of Disconnect 35](#_Toc89863671)

[4.4.2.7 SUN FLP Replenishment 1](#_Toc89863672)

[4.4.2.8 PLP Fixed Wireless 3](#_Toc89863673)

[4.4.2.9 Home Wifi 4](#_Toc89863674)

[4.4.3 Intra MNP 1](#_Toc89863675)

[4.4.3.1 Postpaid to Prepaid 1](#_Toc89863676)

[4.4.3.2 Postpaid to Postpaid 4](#_Toc89863677)

[4.4.3.3 Prepaid to Postpaid 7](#_Toc89863678)

[4.4.3.4 Prepaid to Prepaid 10](#_Toc89863679)

[4.4.4 Inter MNP 13](#_Toc89863680)

[4.4.4.1 Port Out (Postpaid) 13](#_Toc89863681)

[4.4.4.2 Port Out (Prepaid) 14](#_Toc89863682)

[4.4.4.3 Port In (Postpaid Pre-Activation) 16](#_Toc89863683)

[4.4.4.4 Port In (Postpaid Full Activation) 17](#_Toc89863684)

[4.4.4.5 Change MSISDN (Postpaid) 19](#_Toc89863685)

[4.4.4.6 Port In (Prepaid) 20](#_Toc89863686)

[4.4.5 Inter MNP (Non-DSA) 22](#_Toc89863687)

[4.4.5.1 Port In (Postpaid Full Activation) 22](#_Toc89863688)

[4.5 CDPA - High Level Workflow 24](#_Toc89863689)

[4.5.1 Non CDPA – Optima Notification 28](#_Toc89863690)

[4.5.1.1 Subscriber Tagging 29](#_Toc89863691)

[4.6 Special Handling 29](#_Toc89863692)

[4.6.1 Critical NE handling 30](#_Toc89863693)

[4.6.2 RFS Sorting 30](#_Toc89863694)

[4.6.3 RFS Aggregation 30](#_Toc89863695)

[4.6.4 CRI Inquiry 30](#_Toc89863696)

[4.6.5 ISM Inquiry 31](#_Toc89863697)

[4.6.6 DP9 Permanent Disconnect Special Handling 31](#_Toc89863698)

[4.6.7 Intra MNP 1](#_Toc89863699)

[4.6.8 Inter MNP 1](#_Toc89863700)

[4.7 Response Handling 1](#_Toc89863701)

[4.7.1 Intermediate Notification via handleResponse (command and response format) 1](#_Toc89863702)

[4.7.2 Final Response via handleResponse (command and response format) 2](#_Toc89863703)

[4.7.3 HandleResponse command response format 4](#_Toc89863704)

[4.8 Logic Libraries 4](#_Toc89863705)

[4.8.1 Error/Rollback Handling 5](#_Toc89863706)

[4.8.2 Error Whitelisting 5](#_Toc89863707)

[4.8.3 Rollback Handling 6](#_Toc89863708)

[4.9 Catalog HLD 7](#_Toc89863709)

[4.9.1 Catalog Item 7](#_Toc89863710)

[4.9.2 Resource Facing Services Actions 7](#_Toc89863711)

[4.9.3 Request Types 8](#_Toc89863712)

[4.9.4 Catalog Modelling 8](#_Toc89863713)

[4.9.5 Sample RFS Model 8](#_Toc89863714)

[4.9.6 RFS 9](#_Toc89863715)

[4.9.6.1 DATA 9](#_Toc89863716)

[4.9.6.2 OCS OFFER 9](#_Toc89863717)

[4.9.6.3 MVOICE 10](#_Toc89863718)

[4.9.7 Services to RFS mapping 10](#_Toc89863719)

[4.9.8 Undo Transactions 12](#_Toc89863720)

[5 Non-Functional Requirement 12](#_Toc89863721)

[6 Appendix A (RFS Catalog) 13](#_Toc89863722)

[6.1 RFS Catalog 13](#_Toc89863723)

[7 Appendix B (Reference Documents) 14](#_Toc89863724)

[8 Appendix C (Document Change Tracker) 14](#_Toc89863725)

[9 Appendix D (Change Tracker) 1](#_Toc89863726)

**[10](#_Toc89863727)** [HLD Acceptance 1](#_Toc89863727)

Figures:

[Figure 1. Network Context Overview 8](#_Toc49964503)

[Figure 2. FlowOne Order Process Flow 10](#_Toc49964504)

[Figure 3. Request Acknowledgement Handling 16](#_Toc49964505)

[Figure 4. Nokia Service Catalog Life Cycle Management 20](#_Toc49964506)

[Figure 5. Target System Architecture 22](#_Toc49964507)

[Figure 6. Target Application Architecture 23](#_Toc49964508)

[Figure 7. Process Workflow 23](#_Toc49964509)

[Figure 8. Multiple line items processing flow 21](#_Toc49964510)

[Figure 9. Single line item processing flow 22](#_Toc49964511)

[Figure 10. Sample RFS Model 8](#_Toc49964512)

[Figure 11. Undo Transaction 12](#_Toc49964513)

Tables:

[Table 1. Services supported by target solution 6](#_Toc62228123)

[Table 2. Northbound integrations 13](#_Toc62228124)

[Table 3. East/Westbound Integration 17](#_Toc62228125)

[Table 4. Existing Southbound Integration 17](#_Toc62228126)

[Table 5. New Southbound Integration 18](#_Toc62228127)

[Table 6. New Southbound Integration for Inter MNP 19](#_Toc62228128)

[Table 7. Generic BST framework extension 24](#_Toc62228129)

[Table 8. CRI Inquire Value Mapping 26](#_Toc62228130)

[Table 9. Logic Libraries 4](#_Toc62228131)

[Table 10. Phase 1a - Service to RFS Mapping 10](#_Toc62228132)

[Table 11. Phase 1b - Service to RFS Mapping 11](#_Toc62228133)

[Table 12. Intra MNP – Service to RFS Mapping 11](#_Toc62228134)

[Table 13. Inter MNP – Service to RFS Mapping 11](#_Toc62228135)

[Table 14. Fixed Wireless – Additional RFS 11](#_Toc62228136)

# About This Document

This document is a high-level design document. It is aimed to describe the Catalog Driven Provisioning and Activation solution for Trident Optima Integration at a high level. Any low level technical detail will be defined in LLD (Function Specifications Document) at a later stage.

The Optima-USPS project has two Phases, which cover the following:

* Phase Ia
  + Basic services for wireless prepaid and postpaid
    - Includes RFS related to Critical NEs i.e. HLR, UPCC, ONEOCS, DP9
  + CRI Inquiry Special Handling
  + Optima Notification for DSA
* Phase Ib
  + Rest of wireless services for prepaid and postpaid
    - Includes mostly RFS for Non Critical NEs. e.g. CWS
    - Change Bill Cycle
    - Change Ownership
  + Rest of the Optima Notification for non Optima northbounds
    - VASPROV Feature Activation and Deactivation
  + Subscriber Tagging – Optima File Notification
  + Special Handling
    - DP9 Permanent Disconnect Special Handling
  + Notice of Disconnect for VOICE
  + Notice of Disconnect for DATA
  + SAT TV
  + IPTV
* Intra MNP
  + Postpaid to Prepaid
    - Deactivation Postpaid
    - Activation Prepaid
  + Postpaid to Postpaid
    - Deactivation Postpaid
    - Activation Postpaid
  + Prepaid to Postpaid
    - Deactivation Prepaid
    - Activation Postpaid (Will use the existing workflow)
  + Prepaid to Prepaid
    - Deactivation Prepaid
    - Activation Prepaid
* Inter MNP
  + Port out (Postpaid)
    - Deactivation Postpaid
  + Port out (Prepaid)
    - Deactivation Prepaid
  + Port in (Postpaid Pre-Activation)
    - Pre-Activation Postpaid
  + Port in (Postpaid Activation)
    - Activation Postpaid
  + Port in (Prepaid Activation)
    - Activation Postpaid
* Inter MNP (Non-DSA)
  + Port in (Postpaid Activation)
    - Activation Postpaid

The document is describing Phase Ia and Phase Ib for both wireless and wireline services. It is a single document covering the full project. The document will also describe the new added use cases for Intra MNP and Inter MNP.

## Audience

This document is intended to be read by customer staff with a detailed understanding of the technical domain in question, and the ability to identify incorrect, misstated or missing statements and empowered to comment and agree these, as appropriate prior to formal acceptance.

## Terms and Concepts

Following abbreviations, terms and concepts are used in the document:

## Typographic Conventions

The following text styles identify special information used in the document:

|  |  |
| --- | --- |
| *Italics* | Italicized text is used to call attention to cross-references. |
| **Bold** | Bold text is used for presenting:   * field names * menu item names * page names * API names * webservice names   ***Note*** *Notes are written between two lines to call attention to important issues.* |
| Courier | Courier font is used for presenting:   * user input in, for example, commands, parameters and field values * messages shown to the user, file names, scripts and elements |

### Abbreviations

|  |  |
| --- | --- |
| BST | Business Service Tool |
| CRM | Customer Relationship Management |
| CSP | Customer Solution Platform |
| CFS | Customer Facing Services |
| EAI | Enterprise Application Integration |
| ESB | Enterprise Service Bus |
| GUI | Graphic User Interface |
| HLD | High-Level Design |
| HLR | Home Location Register |
| IL | InstantLink (FlowOne Provisioning Activation) |
| LLD | Low-Level Design |
| M/O/C | Mandatory/Optional/Conditional parameter |
| NEI | Network Element Interface |
| OSS/BSS | Operations and Business Support System |
| P&A | Provisioning & Activation |
| SOA | Service Oriented Architecture |
| SOAP | Simple Object Access Protocol |
| MNP | Mobile Number Portability |

### Terminology

|  |  |
| --- | --- |
| network element; NE | Network devices that control network operations, including switching and transport. Example network elements include HLR, GPRS Charging Gateway, MSC and SMSC. |

## Related Documentation

* Nokia FlowOne USPS TSD for Trident SOM & Optima Integration v2.1
* Nokia FlowOne USPS Statement of Work for Trident SOM & Optima Integration v2.4
* PLDT FlowOne19 Installation High Level Design v1.0

**Products document**

* Comptel InstantLink Functional Description
* InstantLink Business Service Tool Functional Description
* InstantLink Java Macro Server Functional Description
* InstantLink Logic Testing Tool Functional Description
* InstantLink Expect Macro Server Functional Description
* Comptel Catalog Functional Description
* InstantLink Batch API Functional Description
* InstantLink SOA Web Services Functional Description
* InstantLink Provisioning Client Functional Description

# Executive Summary

The purpose of this document is to explain the high-level design architecture of Nokia FlowOne Catalog Driven P&A in Trident Optima project on Amdocs wireless services.

This serves as a record of our architecture design discussion and is designed to promote discussion and ensure clarity of communication between Nokia and Amdocs.

The solution offers the benefits that diminish the culminated data gap in the network element manager and the Operating Support Systems (OSS). The solution uses industry standard data modelling for numerous Next Generation Network technologies and the offered solution is agnostic to network technology and vendor.

## Success Factors

To successfully deliver this project following are the key success factors -

* Availability of Service details (Specification documents), in the absence of the document, Amdocs team to help clarifying details of such services, listed below:
  + API specifications for new Northbound Integrations mentioned in Section 3.4.1
  + API specifications for all new Southbound Integrations mentioned in Section 3.4.2
* Hardware readiness (Production, Test& Development).
* Signoff of HLD, LLD in time bound manner.
* To provide all data needed to support the analysis, design, implementation and testing in a timely manner, namely:
  + Trident RFS command mapping with provisioning and rollback commands and any dependencies
  + Test data for our system and non functional testing
  + All the relevant special handling required for implementation

# Overview

This chapter will cover the design overview of the solution. Details are covered in the following chapters or relevant LLDs, this is only for introduction and summary.

## Services

Table 1. Services supported by target solution

|  |  |  |
| --- | --- | --- |
| Type | Use Case | Description |
| Wireless | | |
|  | Activation | Activate Wireless Subscriber |
|  | Change | Modification of Wireless Subscriber   * Change MSISDN/MSISDN Swap * Change SIM/Sim Swap * Add Optional Add-On * Disconnect Optional Add-On * Add Commercial Device * Brand Migration * Change Plan |
|  | Change Bill Cycle | Change Bill cycle |
|  | Change Ownership | Change of Ownership |
|  | Bar/Unbar | Bar/Unbar services and features |
|  | Suspend/Resume | Suspend/resume services (billing suspension) |
|  | Disconnect | Disconnect service |
| IPTV | Activation |  |
|  | Activate PPV |  |
|  | Disconnect PPV |  |
|  | Change Ownership |  |
| SAT TV | Activation SAT TV |  |
|  | Deactivation SAT TV |  |
|  | Change Plan |  |
|  | Change Ownership |  |
|  | Activate PPV |  |
|  | Deactivate PPV |  |
|  | Activate Theme Pack |  |
|  | Deactivate Theme Pack |  |
|  | Suspend SAT TV |  |
|  | Resume SAT TV |  |
| ND | Notice of Disconnect |  |
| DSA | Optima Notification |  |
| FMS | Optima Notification |  |
| VASPROV | Optima Notification |  |
| PLP Fixed Wireless | Activate Wireless Subscriber |  |
|  | Modify Wireless Subscriber |  |
|  | Suspend/Resume |  |
|  | Disconnect |  |
| Intra MNP | | |
|  | Postpaid to Prepaid |  |
|  | Postpaid to Postpaid |  |
|  | Prepaid to Postpaid |  |
|  | Prepaid to Prepaid |  |
| Inter MNP | | |
|  | Port Out (Postpaid & Prepaid Deactivation) |  |
|  | Port In (Postpaid Pre-Activation) |  |
|  | Port In (Postpaid Activation) |  |
|  | Port In (Prepaid Activation) |  |
|  | Change MSISDN of Wireless Service (Postpaid) |  |
| Inter MNP (Non-DSA) | | |
|  | Port In (Postpaid Activation) |  |

## Network Context



Figure 1. Network Context Overview

Below are the Network Elements supported for each Phase.

**Phase 1a Network Elements**

* + Nokia OneNDS HLR
  + Huawei ONEOCS
  + Huawei UPCC
  + DP9 SOAP
  + Optima SOAP - (Northbound notification)
  + CRI DB Cache

**Phase 1b Network Elements**

* + APTILO
  + ATS
  + CDP XML over HTTP
  + CIGNAL
  + CWS SOAP
  + CWS File-based
  + ENUM
  + FMS DB
  + ISM
  + MYSMART File-based
  + SERVICE DESQ
  + SSP SEP proprietary
  + U2000
  + UREG

**Intra MNP**

New NE

* + AGW
  + MPCC
  + OTA

**Inter MNP**

New NE

* + DSA

Additional Command to existing NEI:

* + ISM (portInAct)

All the network capabilities will be available in Nokia Service Catalog as technical services. Provisioning workflow does not need to be changed when changing the RFS definition

## Processes

This section highlights the order process flow within the Nokia FlowOne provisioning and activation system, the orders request starting from CRM and up to the delivery to network. This section highlights the role of Nokia’s CDPA architecture-based flow design.

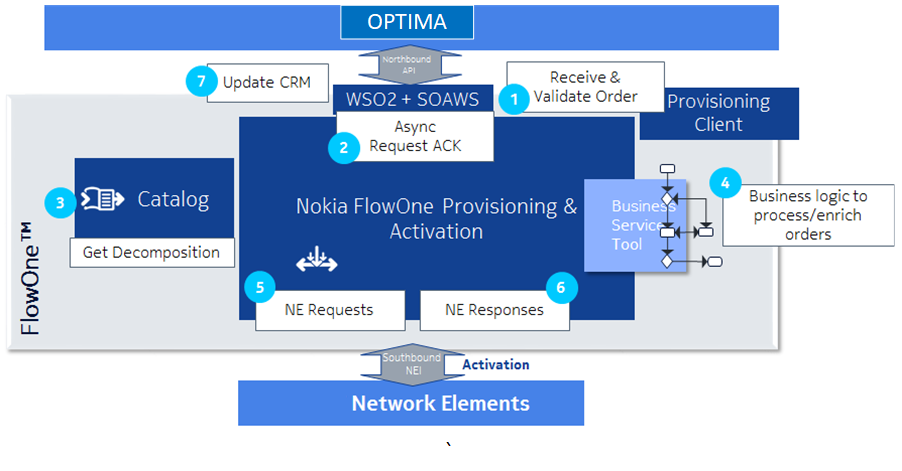


Figure 2. FlowOne Order Process Flow

After an order is submitted to Nokia FlowOne for processing, the following steps describe the technical process of Catalog-driven provisioning and activation for an order delivery example across the different systems:

1. Nokia FlowOne P&A for Optima-USPS environment supports requests from Amdocs Optima Ordering and Service Activation. In the above diagram, the order receives through a Northbound **SOA web service API**. The order is assigned to Main BST workflow, where Nokia FlowOne P&A perform an order line item validation check against the schema definition and order data mapping against the lookup table.
2. There is a transformation proposed via a northbound adapter **WSO2**1 enterprise service bus platform to Nokia FlowOne SOA compliant format, to cater Optima’s multiple RFS blocks provisioning. (see Appendix C for sample SOAP request)
3. As part of the Optima design requirement, the northbound system expects the request acknowledgement notification after the requests being validated based on expected format. **WSO2**1 ESB will handle acknowledgement notification towards Optima northbound.
4. Nokia FlowOne P&A decomposes one RFS into lower level Technical Service and creates a provisioning plan-based Task Definitions on **Nokia Service Catalog**.
5. **Business Logic Tool module (BST)** then perform business rules validation/processing and order enrichment. This include *network element2* queries and activations to fulfil the services.
6. Nokia FlowOne P&A integrate with Amdocs network element for *CRUD3* process via **Network Element Interface module (NEI)** module. Decomposed lower level service (RFS) will populate a lists of technical services (TS) to be sent over as task requests via designated format and protocol as required by each NE.
7. NEI receive the responses from network element and will be processed by BST, in which the responses will be mapped based on lookup table or any special handling based on the workflow.
8. Nokia will proceed with the next RFS and repeat steps 4-7, until all RFS are executed.
9. Nokia FlowOne P&A then completes the order and send the final response to Amdocs Optima.

*Note WSO21 – WSO2 Enterprise Service Bus is an open source product with lightweight, high performance and comprehensive ESB.*

*Note network element2 – Network element supported is listed in section Southbound Interfaces*

*Note CRUD3 – Create, read, update and delete operation that supported by network element.*

### User Interfaces

Default (OOB) GUIs will be used as user interface to different applications. All user interfaces will be in English and no new/customized user interfaces for the following:

* Nokia FlowOne Provisioning & Activation User Interface
* Nokia Service Catalog Designer GUI
* Nokia Provisioning Client GUI1

*Note 1Customized forms in the default Nokia Provisioning Client GUI will be implemented.*

### Reporting

The following standard reports will be provided during this phase:

* Database Usage Report
* Cleaned Requests
* System Configuration Reports
* Database GRC Parameter Report
* Network Model Report
* Service Module Report
* User Information Report
* Request Information Reports
* Request Summary Report
* Task Summary Report
* Task summary report for defined NE type
* Task summary report for defined NE
* Request Details Report (per one request)

*Note The above reports will be in English and are available as standard product features and are noted here for ease of reference.*

### Customized Reporting

An hourly customized report will be implemented as a standalone script that will provide customers needed information for their fallout handling/separate reporting. This is based on available provisioning parameters from IL database. Below is the expected filename format and content based on legacy USPS reports requirement.

#### IRAFM Reports

A report that will display task level information of requests generated per hour.

Filename format: PROV\_USPS1\_<yyyymmddhh>.tmp

File Content:

IMSI!@#MSISDN!@#NE\_ID!@#TECHNICALPRODUCT!@#ACTIONCODE!@#SRVC\_CODE!@#NETWORK\_ELEMENT!@#SO\_RCVDATE!@#CP\_PROCDATE!@#MML\_SEQUENCE!@#STATUS\_ID!@#CLIENT\_PROCDATE!@#CP\_RCVDATE!@#OLD\_MSISDN!@#OLD\_IMSI!@#SOURCE\_NAME!@#DTS\_CODE!@#REMARKS

515030116661308!@#639476518656!@#cri!@#MVOICE!@#ADD!@#null!@#CRI!@#30-APR-20 02.07.10.230 PM!@#30-APR-20 02.07.30.279 PM!@#null!@#7!@#30-APR-20 02.07.10.266 PM!@#30-APR-20 02.07.30.279 PM!@#null!@#null!@#TAF!@#null!@#null

515030116661308!@#639476518656!@#pgw1!@#MVOICE!@#ADD!@#null!@#HLR!@#30-APR-20 02.07.10.230 PM!@#30-APR-20 02.07.30.279 PM!@#null!@#7!@#30-APR-20 02.07.10.266 PM!@#30-APR-20 02.07.30.279 PM!@#null!@#null!@#TAF!@#null!@#null

#### IRAFM Manifest File

A manifest file is created per IRAFM report generated.

Filename format: MANIFEST\_<Filename of file sent>\_YYYYMMDD.txt

File Content:

DataFileName|FileCreationTimestamp|FileClosureTimestamp|OpeningSequenceNumber|ClosingSequenceNumber|ControlTotal|FileSize|FileChecksum

PROV\_USPS1\_2019121912.tmp|20191219141844||||51879|12365877|4de1ad665444eb8495aae4d321937257

### Localization

No requirement on different locales required in terms of the combinations of language, layout, character set, etc.

### FlowOne (InstantLink) Archive

FlowOne orders that received over the defined interval will be transferred to Archive Server and the frequency of archive orders from FlowOne is recommended as 4 hours and the value is configurable via GUI. Archived orders can be viewed through the same FlowOne GUI.

## Integrations

### Northbound Interfaces

Table 2. Northbound integrations lists the northbound integrations that are in the scope.

Table 2. Northbound integrations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Source System | Description of Source System | Target System | Target Integration | Protocol |
| 1 | OPTIMA | Amdocs BIL | FlowOne P&A | XML (REST) | HTTPS |

#### Protocol

Nokia SOA WS API will be exposed to northbound interface via WSO2. SOA WS API support following:

* WSDL version 1.1
* SOAP version 1.2
* HTTPS as the standard protocol (WSO2)

#### Message Structure

In order to support multiple RFS request structure from Amdocs BIL/Optima northbound, a message converter will be done by FlowOne via WSO2. In general, FlowOne supports in total of two type of SOA messages that will be sent by northbound system (converted by WSO2).

* Asynchronous SOAP Messages
* Synchronous SOAP Messages

##### Asynchronous SOAP Messages

All use cases send via this method.

All Use-Cases request message format for structure is as below and will be converted to SOA WS native format by WSO2:

**Example: Asynchronous request message:**

<ModifyRequest>

<RequestHeader>

<NeType>OPTIMA</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>OPTIMA</ReqUser>

<RequestHeader>

<RequestParameters>

<Parameter name=“account\_number" value=“28292020"/>

<RFS>

<Parameter name=“rfs" value=“IPTV"/>

<Parameter name=“action" value=“REMOVE"/>

<Parameter name=“contractno" value=“contract#1"/>

<Parameter name=“plancode" value=“PLDT4999"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“IPTV"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“contractno" value=“contract#1"/>

<Parameter name=“plancode" value=“PLDT5999"/>

</RFS>

</RequestParameters>

</ModifyRequest>

All Use-Cases request acknowledgement message format and response format will follow SOA standard acknowledgement and response structure.

**Example: Acknowledgement message:**

<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">

<S:Body>

<Response xmlns="http://soa.comptel.com/2011/02/instantlink">

<ResponseHeader> 

<RequestId>351</RequestId>

<Status>Message accepted</Status> 

<OrderNo>123456</OrderNo>

<StatusMessage>InstantLink accepted request with request id: 351

for order no: 123456</StatusMessage>

</ResponseHeader>

<ResponseParameters/> 

</Response>

</S:Body>

</S:Envelope>

##### Asynchronous Request Acknowledgement Handling



Figure 3. Request Acknowledgement Handling

Asynchronous requests sent by northbound systems should be translated/converted into Nokia SOA API format before being directed to FlowOne.

After FlowOne receiving the Asynchronous request, a request acknowledgement will be returned to northbound to notify that the request has been accepted.

Upon request processing is completed, response will be generated and responded to the northbound system if the reply-to-address is included in the callback tag. In scenario where requester is not OPTIMA, additional SOAP response will always be required to send to OPTIMA for status update purpose.

**Note**: Final response will be sent back to Optima via an NEI, thus, the replyTo address will be defined as fixed value pointing to InstantLink SOA Web Services’ Response WSDL mock service. This is to ensure all requests will be set to ACK after processing.

#### Batch API

Batch API is used for bulk request processing. Following use cases supports provisioning via default batch api format:

* Change Bill Cycle for Wireless Subscriber and Wireline
* Notice for Disconnect

Default batch api format follows a standard filename and key-value pair content format that shall follow by northbound sending the file. By utilizing the standard api, there is no need to implement new batch plugin scripts.

### East/West Interfaces

Table 3. East/Westbound Integration lists and describes the east/west bound integrations that are in the scope of project.

Table 3. East/Westbound Integration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Source System** | **Source Integration** | **Target System** | **Description** | **Protocol** |
| 1 | FlowOne P&A | FlowOne Database | FlowOne Archive Database | Nokia FlowOne (InstantLink) Archive | TCP |

Nokia FlowOne (InstantLink) Archive can be equipped with a functionality which enables the transfer of requests and tasks data to a FlowOne (InstantLink) Archive database. This data can be used for making historical analysis and generating customer specific reports using external tools.

**NOTE:** Data in archive database will only be kept on the configured retention period.

### Southbound Interfaces

lists and describes the southbound integrations that are in the scope of the project

Table 4. Existing Southbound Integration

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Source System | Source Integration | Target System | Protocol |
| 1 | FlowOne P&A | NEI | APTILO | REST |
| 2 | FlowOne P&A | NEI | ATS | SOAP |
| 3 | FlowOne P&A | NEI | CDP | XML over HTTP |
| 4 | FlowOne P&A | NEI | CRI | Java API |
| 5 | FlowOne P&A | NEI | CWS | FTP (File) |
| 6 | FlowOne P&A | NEI | CWS | SOAP |
| 7 | FlowOne P&A | NEI | DP9 | SOAP |
| 8 | FlowOne P&A | NEI | ENUM | SOAP |
| 9 | FlowOne P&A | NEI | FMS | DB Connection |
| 10 | FlowOne P&A | NEI | ISM | SOAP |
| 11 | FlowOne P&A | NEI | MYSMART | FTP (File) |
| 12 | FlowOne P&A | NEI | ONEOCS | WS |
| 13 | FlowOne P&A | NEI | OneNDS (NTHLR) | SPML |
| 14 | FlowOne P&A | NEI | OPTIMA | REST (JSON) |
| 15 | FlowOne P&A | NEI | SERVICE DESQ | WS |
| 16 | FlowOne P&A | NEI | SSP | SEP proprietary |
| 17 | FlowOne P&A | NEI | UPCC | WS |
| 18 | FlowOne P&A | NEI | U2000 | Telnet |
| 19 | FlowOne P&A | NEI | UREG | SOAP |

Table 5. New Southbound Integration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Source System | Source Integration | Target System | Protocol | Operation Type |
| 1 | FlowOne P&A | NEI | OPTIMA | REST (JSON) | C,R,U,D |
| 2 | FlowOne P&A | NEI | CIGNAL | WS | C,R,U,D |
| 3 | FlowOne P&A | NEI | AGW | SOAP | C,R,U,D |
| 4 | FlowOne P&A | NEI | MPCC | SFTP/FTP (File based) | C,R,U,D |
| 5 | FlowOne P&A | NEI | OTA | HTTP | C,R,U,D |

Table 6. New and Existing Southbound Integration for Inter MNP

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Source System | Source Integration | Target System | Protocol | Operation Type |
| 1 | FlowOne P&A | NEI | DSA | SOAP | C |
| 2 | FlowOne P&A | NEI | ISM (Existing NEI) | SOAP | (New command portInAct() to be added) |

### Nokia Service Catalog

The FlowOne Business Service Tool (BST) interface for Nokia Service Catalog is an internal interface of Nokia Service Catalog that allows it to be integrated with Nokia FlowOne. It is an add-on component of FlowOne BST that can be used to access Nokia Service Catalog from the provisioning logic.

The interface is used to get product and service specifications and decompositions from Nokia Service Catalog. When the decomposition is centrally available in Nokia Service Catalog, BST logic will become smaller and easier to manage.

During a provisioning request, an internal interface is used between FlowOne BST and Nokia Service Catalog to fetch the decomposition data. This interface is responsible for fetching the product or service decomposition and for populating any parameters defined in the low-level items with the equivalent values supplied as part of the provisioning request.

Nokia Service Catalog consists of CFS and RFS items and CFS are the top layer item in Catalog. As part of the Trident Optima requirements to model catalog items as RFS, so this RFS will be send to CDPA solution. Below diagram specifies the various life cycle states of an CFS/RFS in Catalog

This section defines the CFS/RFS life cycle of items within Service Catalog.



Figure 4. Nokia Service Catalog Life Cycle Management

* **Working** – A new version has been generated, either on the basis of the selected version object or from scratch. This is the only state in which the object can be modified freely. However, it can not be referenced by external systems.
* **Testing** – A version is used for testing. It is not possible to edit a version in the Testing state. However, such a version can be accessed by an external system, such as, Nokia FlowOne for testing purposes. If the tests fail, the version can be returned back to the Working state for further editing. Versions that pass the test can be published.
* **Published** – A version of a product, service, technical library or technical service has been released for production use and can be referenced by external systems. It is not possible to make changes to a Published version.
* **Suspended** – When a version of a product, service, technical library or technical service has been suspended, it cannot be accessed by external systems. The suspended product, service, technical library or technical service can either be reverted to the Published state or it can be set as Outdated.
* **Outdated** – There are still subscriptions to a product, service, technical library or technical service version, but the version cannot be used for new subscriptions.
* **Terminated** – A version is not accessible by external systems anymore. At this stage there is no need for any other system to refer to this version, as there are no subscriptions to this version. Terminated versions can be removed from Nokia Service Catalog.

# Solution Architecture

## Product

Following Nokia products will be introduced in the target solution:

**Nokia FlowOne Provisioning & Activation:** Provides business logic configuration and external integration capabilities including northbound, southbound and also west-/eastbound integrations via FlowOne platform.

**Nokia Service Catalog:** Nokia Service Catalog allows the customer to create and define products and services breakdown and associated rules and dependencies using Service Catalog GUI.

In this delivery, provisioning and activation would be catalog driven (**CDPA**), which works independently of workflow design, maintained in existing BST logics. Nokia Service Catalog breaks down the services to lower level resource facing services (**RFS**). Nokia FlowOne P&A workflow can use them during the run-time provisioning process by querying the different items from Nokia Service Catalog.

**Event Management:** Functions as a mediator between Nokia FlowOne written applications and the customer’s alarm monitoring system.

**Nokia Provisioning and Activation (IL) Archive:** Enables Nokia Provisioning and Activation (InstantLink) to transfer request and task data, as well as statistical data about their processing, to a (IL) Archive database. This data can be used for making historical analysis and generating customer specific reports.

## Target Solution Architecture

The diagram below illustrates the target system architecture solution, with details of the key scope objectives and the interaction points with external systems including the various applications which are part of the proposed solution.

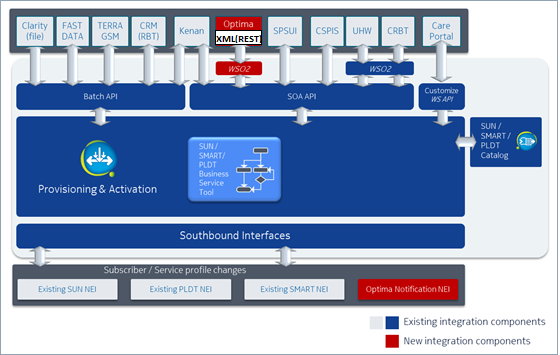


Figure 5. Target System Architecture

**Note**: Only requests coming from Optima will be covered in this new RFS-model solution. All other northbound integrations will be retained in Legacy USPS. Optima Notification will also be introduced and implemented in the Legacy USPS solution.

## Target Application Architecture

In the current solution in USPS, services are modelled in the form of product name, landline/product type and SNE and technical services in the Nokia Service Catalog. Moving forward in the target solution, new Resource Facing Services (RFS) and existing technical services need to be modelled and managed in the Nokia Service Catalog. RFS represent the network element and its subscription/feature whilst Technical Services represent the technical action/commands towards the network elements.



Figure 6. Target Application Architecture

## High Level Process Workflow



Figure 7. Process Workflow

1. Once the service order is received by FlowOne, an Acknowledgement response is sent back to Amdocs BIL./Optima.
2. RFS will be sorted based on predefined RFS sequencing configured in the lookup table.
3. All RFS related to Critical NEs are provisioned prior to Non Critical RFSs.
4. USPS triggers a request to the Network Elements to activate the service, and receives provisioning response from NEs
   1. If Critical RFS(s) provisioning is successful, USPS will send a Success notification to Amdocs BIL and continue provisioning the Non Critical RFS(s).
   2. If Critical RFS provisioning fails, then it will return a Failure message to Amdocs Ordering via BIL, and will not continue with Non Critical RFS provisioning. Rollback will be executed for all successful Critical RFS(s) that are eligible for rollback. (*see Section 4.8.3 for Rollback Eligibility*)
   3. If Non-Critical NE provisioning is failed, then final status will be failure response to Amdocs BIL.
5. Final response will be sent to Amdocs BIL for the overall status of provisioning.

**Note**: Intermediate notification will be handled via the same NEI command that handles final response.

### Phase 1a

The requests format below will be transformed via WSO2 layer to follow the standard SOA format of FlowOne. The final format of the requests will be finalized in the LLD stage, below are just sample to show the RFS(s) that goes with each use cases.

#### Activation of Wireless Subscriber

Amdocs BIL sends request to activate a service on the network elements.

##### Example

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle" value=“M01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name="“rfs" value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“CUG"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“cug\_member1" value=“9988453333"/>

<Parameter name=“cug\_member2" value=“9988454444"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### Wireless Service Modification

Amdocs BIL sends request to modify a service on the network elements.

##### Example

<ModifyRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle" value=“M01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER”/>

<Parameter name=“action" value=“ADD"/>

<Parameter name="ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</ModifyRequest>

#### Suspend/Resume Wireless Subscriber

Amdocs BIL sends request to suspend/resume a service on the network elements.

##### Example (Suspend)

<ModifyRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle" value=“M01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“MVOICE"/>

<Parameter name=“action" value=“SUSPEND"/>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“SUSPEND"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“ MOBILEINTERNET"/>

<Parameter name=“action" value=“SUSPEND"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“SUSPEND"/>

</RFS>

</RFS>

</RequestParameters>

</ModifyRequest>

##### Example (Resume)

<ModifyRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle" value=“M01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“MVOICE"/>

<Parameter name=“action" value=“RESUME"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“RESUME"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“MOBILEINTERNET"/>

<Parameter name=“action" value=“RESUME"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“RESUME"/>

</RFS>

</RequestParameters>

</ModifyRequest>

#### Disconnect Wireless Subscriber

Amdocs BIL sends request to disconnect a service on the network elements.

##### Example

<DeleteRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle" value=“M01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“MVOICE"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

</RequestParameters>

</DeleteRequest>

### Phase 1b

The requests format below will be transformed via WSO2 layer to follow the standard SOA format of FlowOne. The final format of the requests will be finalized in the LLD stage, below are just sample to show the RFS(s) that goes with each use cases.

#### Change Billing Cycle

Amdocs BIL sends request to trigger change billing cycle on the network elements. This is supported via file based requests and will be following standard batch api format.

##### Batch Sample Input File

Optima will dump a batch input file to USPS server for provisioning. For wireline, there is additional need to inquire from ARM to retrieve OPTION82 and provision commands to PCRF.

Sample file format for Wireless:

| File Format : | BwsCHGBC\_20170814000051.req |
| --- | --- |
| Content: | NE\_TYPE ORDER,NE\_ID hlr1,REQ\_TYPE 2,REQ\_USER Optima,ORDERTYPE Modify,IMSI 99999988453333,MSISDN 9988453333,ACCOUNT\_NUMBER 435455,ACCOUNT\_TYPE Postpaid, BILL\_CYCLE 1,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT MVOICE,SO1\_ACTIONCODE CHGBILLCYCLE |

Sample file format for Wireline:

| File Format : | wireline\_usps\_billcycle\_sysdate.req |
| --- | --- |
| Content: | NE\_TYPE ORDER,NE\_ID hlr1,REQ\_TYPE 2,REQ\_USER Optima,ORDERTYPE Modify,ACCOUNT\_NUMBER 435455,ACCOUNT\_TYPE Postpaid, BILL\_CYCLE 1,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT DSL,SO1\_ACTIONCODE CHGBILLCYCLE, SO1\_ACCESS\_ID TMC1285-042018-59992 |

#### Change Ownership

Amdocs BIL sends request to move service from one account to another account.

##### Example

<ModifyRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle" value=“M01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“MVOICE"/>

<Parameter name=“action" value=“CHGOWNERSHIP"/>

<Parameter name="old\_account\_number" value=“87667501"/>

<RFS>

</RequestParameters>

#### Bar/Unbar Wireless Subscriber

Amdocs BIL sends request to a bar/unbar services.

##### Example

<ModifyRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle" value=“M01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“BARINCOMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“BAROUTGOING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

</RequestParameters>

Below wireline services will be supported in as well in Optima:

#### IPTV

Amdocs BIL sends request to create iptv customer, activate pay per view, deactivate PPV service and change ownership on the network elements.

##### Example

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“account\_number" value=“28292020"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“IPTVCUSTOMER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“operationalentity" value=“T01562"/>

<Parameter name=“category" value=“POSTPAID"/>

<Parameter name=“customertype" value=“3PLAY"/>

<Parameter name=“salescode" value=“MSI000"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### SAT TV

Amdocs BIL sends request to activate, modify or deactivate SAT TV service or PPV service on the network elements.

##### Example

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>Optima</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“account\_number" value=“28292020"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs" value=“SATTVCUSTOMER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“operationalentity" value=“T01562"/>

<Parameter name=“category" value=“POSTPAID"/>

<Parameter name=“customertype" value=“3PLAY"/>

<Parameter name=“salescode" value=“MSI000"/>

</RFS>

</RequestParameters>

#### Notice of Disconnect

Notice of disconnect is triggered via batch input file that will contain the list of subscribers that are due for barring and disconnection. Each entry from the input file will be provisioned as one request going to FlowOne.

USPS needs to inquire from ARM to retrieve OPTION82 and provision commands to network elements.

##### (Batch input format)

Redirection:

| File Format : | HOME\_ND\_VOICE\_20200211093337.req |
| --- | --- |
| Content: | NE\_TYPE ORDER,REQ\_TYPE 2,REQ\_USER Optima,ORDERTYPE Modify,ACCOUNT\_NUMBER 435455, REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT VOIP,SO1\_ACTIONCODE ND, SO1\_BALANCE\_DUE 742,SO1\_BALANCE\_DUE\_DATE 27-JAN-20,SO1\_TELEPHONE 0629451641 |

Lifting:

| File Format : | HOME\_ND\_VOICE\_LIFTED\_20200127100629.req |
| --- | --- |
| Content: | NE\_TYPE ORDER,REQ\_TYPE 2,REQ\_USER Optima,ORDERTYPE Modify,ACCOUNT\_NUMBER 435455,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT VOIP,SO1\_ACTIONCODE NDLIFT,SO1\_TELEPHONE 0286462771 |

Redirection:

| File Format: | **HOME\_ND\_DATA\_YYYYMMDDHHMISS.req**  HOME\_ND\_DATA\_20200211093339.req |
| --- | --- |
| Content: | NE\_TYPE ORDER,REQ\_TYPE 2,REQ\_USER Optima,ORDERTYPE Modify,ACCOUNT\_NUMBER 435455,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT DSL,SO1\_ACTIONCODE ND,SO1\_SERVICE\_ID 0459827293\_DSL,SO1\_BALANCE\_DUE 4046.11,SO1\_BALANCE\_DUE\_DATE 27-JAN-20, SO1\_ACCESS\_ID TMC1285-042018-59992 |

Lifting:

| File Format: | **HOME\_ND\_DATA\_LIFTED\_YYYYMMDDHHMISS.req**  HOME\_ND\_DATA\_LIFTED\_20200127100629.req |
| --- | --- |
| Content: | NE\_TYPE ORDER,REQ\_TYPE 2,REQ\_USER Optima,ORDERTYPE Modify,ACCOUNT\_NUMBER 435455,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT DSL,SO1\_ACTIONCODE NDLIFT,SO1\_ SERVICE \_ID 026462771\_DSL, SO1\_ACCESS\_ID TMC1285-042018-59992 |

#### 

#### SUN FLP Replenishment

The Amdocs Billing extract utility will send an FLP replenishment request to USPS for provisioning of Add-on replenishments (MINs). Each entry from the input file will be provisioned as one request going to FlowOne.

##### (Batch input format)

| File Format : | | KENAN\_REP\_FLP\_ANNIV\_YYYYMMDDHHMMSS.req |
| --- | --- | --- |
| Content: | | NE\_TYPE ORDER,REQ\_TYPE 1,REQ\_USER Optima,ORDERTYPE New,IMSI 99999988453333,MSISDN 639988453333,ACCOUNT\_NUMBER 435455,ACCOUNT\_TYPE Postpaid, BILL\_CYCLE 30,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT MVOICE,SO1\_ACTIONCODE ACTCREDIT,SO1\_AMOUNT 344,SO1\_EXPIRY\_DAYS 30,SO1\_IN\_SERVICE\_ACCOUNT FLP300 |
| File Format : | KENAN\_REP\_FLP\_DAILY\_YYYYMMDDHHMMSS.req | |
| Content: | NE\_TYPE ORDER,REQ\_TYPE 1,REQ\_USER Optima,ORDERTYPE New,IMSI 99999988453333,MSISDN 639988452222,ACCOUNT\_NUMBER 435455,ACCOUNT\_TYPE Postpaid, BILL\_CYCLE 30,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT MVOICE,SO1\_ACTIONCODE ACTCREDIT,SO1\_AMOUNT 344,SO1\_EXPIRY\_DAYS 30,SO1\_IN\_SERVICE\_ACCOUNT FLP300 | |
| File Format : | | KENAN\_REP\_FLP\_AUTO\_YYYYMMDDHHMMSS.req |
| Content: | | NE\_TYPE ORDER,REQ\_TYPE 1,REQ\_USER Optima,ORDERTYPE New,IMSI 99999988453333,MSISDN 639988451111,ACCOUNT\_NUMBER 435455,ACCOUNT\_TYPE Postpaid, BILL\_CYCLE 30,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT MVOICE,SO1\_ACTIONCODE ACTCREDIT,SO1\_AMOUNT 344,SO1\_EXPIRY\_DAYS 30,SO1\_IN\_SERVICE\_ACCOUNT FLP300 |

##### (Batch response format)

| File Format : | | KENAN\_REP\_FLP\_ANNIV\_YYYYMMDDHHMMSS.req.ready |
| --- | --- | --- |
| Content: | | STATUS 9,SMESSAGE Request executed successfully.,@NEW@SO1\_SMESSAGE Task successfully executed.,@REQ@REQ\_DATE\_TIME 20191201160545,@NEW@SO2\_SMESSAGE\_ID BST000,ACTIVATION\_DATE 20200826043540,MAX\_TASK\_TIME 60,FINISHED\_DATE 20200826043544,NE\_TYPE ORDER,SMESSAGE\_ID BST000,@REQ@MSISDN 639988453333,SAS\_ACTION RESPONSE,REQ\_USER Optima,@REQ@REQ\_OBJ 1,@NEW@SO1\_SMESSAGE\_ID BST000,@REQ@IMSI 99999988453333,PRIORITY 5,@NEW@SO1\_STATUS 9,@NEW@SO2\_STATUS 9,@REQ@ACCOUNT\_NUMBER 435455,RETRANSMISSION\_FLAG 0,@REQ@SO1\_EXPIRY\_DAYS 30,@NEW@SO3\_STATUS 9,TASK\_NO 9517,RECEIVED\_DATE 20200826043510,@REQ@SO1\_AMOUNT 344,RESP\_QUEUE\_ID BA\_FLP,@REQ@SO1\_ACTIONCODE ACTCREDIT,@REQ@SO1\_IN\_SERVICE\_ACCOUNT FLP300,REQ\_TYPE 1,@REQ@ORDERTYPE New,@NEW@SO3\_SMESSAGE\_ID BST000,@REQ@SO1\_TECHNICALPRODUCT MVOICE,@REQ@ACCOUNT\_TYPE Postpaid,@REQ@BILL\_CYCLE 30,@REQ@BATCH\_API\_LINE\_NO 1,@REQ@BATCH\_API\_FILENAME KENAN\_REP\_FLP\_ANNIV\_20200826100100.req |
| File Format : | KENAN\_REP\_FLP\_DAILY\_YYYYMMDDHHMMSS.req.ready | |
| Content: | STATUS 9,SMESSAGE Request executed successfully.,@NEW@SO1\_SMESSAGE Task successfully executed.,@REQ@REQ\_DATE\_TIME 20191201160545,@NEW@SO2\_SMESSAGE\_ID BST000,ACTIVATION\_DATE 20200826044031,MAX\_TASK\_TIME 60,FINISHED\_DATE 20200826044034,NE\_TYPE ORDER,SMESSAGE\_ID BST000,@REQ@MSISDN 639988451111,SAS\_ACTION RESPONSE,REQ\_USER Optima,@REQ@REQ\_OBJ 1,@NEW@SO1\_SMESSAGE\_ID BST000,@REQ@IMSI 99999988453333,PRIORITY 5,@NEW@SO1\_STATUS 9,@NEW@SO2\_STATUS 9,@REQ@ACCOUNT\_NUMBER 435455,RETRANSMISSION\_FLAG 0,@REQ@SO1\_EXPIRY\_DAYS 30,@NEW@SO3\_STATUS 9,TASK\_NO 9519,RECEIVED\_DATE 20200826043815,@REQ@SO1\_AMOUNT 344,RESP\_QUEUE\_ID BA\_FLP,@REQ@SO1\_ACTIONCODE ACTCREDIT,@REQ@SO1\_IN\_SERVICE\_ACCOUNT FLP300,REQ\_TYPE 1,@REQ@ORDERTYPE New,@NEW@SO3\_SMESSAGE\_ID BST000,@REQ@SO1\_TECHNICALPRODUCT MVOICE,@REQ@ACCOUNT\_TYPE Postpaid,@REQ@BILL\_CYCLE 30,@REQ@BATCH\_API\_LINE\_NO 1,@REQ@BATCH\_API\_FILENAME KENAN\_REP\_FLP\_AUTO\_20200826100100.req | |

| File Format : | KENAN\_REP\_FLP\_AUTO\_YYYYMMDDHHMMSS.req.ready |
| --- | --- |
| Content: | STATUS 9,SMESSAGE Request executed successfully.,@NEW@SO1\_SMESSAGE Task successfully executed.,@REQ@REQ\_DATE\_TIME 20191201160545,@NEW@SO2\_SMESSAGE\_ID BST000,ACTIVATION\_DATE 20200826043820,MAX\_TASK\_TIME 60,FINISHED\_DATE 20200826043824,NE\_TYPE ORDER,SMESSAGE\_ID BST000,@REQ@MSISDN 639988452222,SAS\_ACTION RESPONSE,REQ\_USER Optima,@REQ@REQ\_OBJ 1,@NEW@SO1\_SMESSAGE\_ID BST000,@REQ@IMSI 99999988453333,PRIORITY 5,@NEW@SO1\_STATUS 9,@NEW@SO2\_STATUS 9,@REQ@ACCOUNT\_NUMBER 435455,RETRANSMISSION\_FLAG 0,@REQ@SO1\_EXPIRY\_DAYS 30,@NEW@SO3\_STATUS 9,TASK\_NO 9518,RECEIVED\_DATE 20200826043720,@REQ@SO1\_AMOUNT 344,RESP\_QUEUE\_ID BA\_FLP,@REQ@SO1\_ACTIONCODE ACTCREDIT,@REQ@SO1\_IN\_SERVICE\_ACCOUNT FLP300,REQ\_TYPE 1,@REQ@ORDERTYPE New,@NEW@SO3\_SMESSAGE\_ID BST000,@REQ@SO1\_TECHNICALPRODUCT MVOICE,@REQ@ACCOUNT\_TYPE Postpaid,@REQ@BILL\_CYCLE 30,@REQ@BATCH\_API\_LINE\_NO 1,@REQ@BATCH\_API\_FILENAME KENAN\_REP\_FLP\_DAILY\_20200826100100.req |

#### PLP Fixed Wireless

Amdocs BIL sends request to activate, modify, suspend/resume and disconnect wireless subscriber. The provisioning flows are the same as Wireless flows. Landline number will be sent by Optima in PLP Number parameter, also same value will be passed under the MSISDN parameter. As the parameter is sent under MSISDN, most wireless flow will work as normal.

**<Parameter name=“msisdn" value=“02345456"/>**

**<Parameter name=“plp\_number" value=“02345456"/>**

All flows should include PLP Number parameter for Fixed Wireless.

##### Example

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo </RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“02345456"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<Parameter name=“plp\_number" value=“02345456"/>

<RFS>

<Parameter name="“rfs value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### Home Wifi

Amdocs BIL sends request to activate, modify, suspend/resume and disconnect wireless subscriber. The provisioning flows are the same as Wireless flows. An additional attribute will be sent for Home wifi provisioning requests

**<Parameter name=“is\_home\_wifi" value=“Y"/>**

**Where**:  
Y-Home Wifi

N-Regular Mobile

Note: parameter not existing means Regular Mobile. If set to Y, a list of network elements from Regular Mobile will not be provisioned or skipped for provisioning.

##### Example

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://localhost:44006/ilws-response-mock-impl/ResponseHandlerService?wsdl</ReplyTo </RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“02345456"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<Parameter name=“is\_home\_wifi" value=“Y"/>

<RFS>

<Parameter name="“rfs value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

</RequestParameters>

</CreateRequest>

### Intra MNP

#### Postpaid to Prepaid

Amdocs BIL sends request to deactivate the postpaid service on the network elements and another request to activate the prepaid service.

##### Example (Postpaid Deactivation)

<DeleteRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>7</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“N"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs” value=“MVOICE"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“OCS\_OFFER"/>

<Parameter name=“action" value=“REMOVE"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</DeleteRequest>

##### Example (Prepaid Activation)

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“Y"/>

<Parameter name=“bill\_cycle" value=“0101"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name="rfs" value="MPVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=”rfs” value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“CUG"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ member1" value=“9988453333"/>

..

<Parameter name=“ member5" value=“9988454444"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### Postpaid to Postpaid

Amdocs BIL sends request to deactivate the old postpaid service on the network elements and another request to activate a new different postpaid service.

##### Example (Postpaid Deactivation)

<DeleteRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>7</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“N"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs” value=“MVOICE"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“OCS\_OFFER"/>

<Parameter name=“action" value=“REMOVE"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</DeleteRequest>

##### Example (Postpaid Activation)

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“Y"/>

<Parameter name=“bill\_cycle" value=“0101"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name="rfs" value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=”rfs” value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“CUG"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ member1" value=“9988453333"/>

..

<Parameter name=“ member5" value=“9988454444"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### Prepaid to Postpaid

Amdocs BIL sends request to deactivate the prepaid service on the network elements and another request to activate the postpaid service.

##### Example (Prepaid Deactivation)

<DeleteRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>7</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“N"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs” value=“MPVOICE"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“OCS\_OFFER"/>

<Parameter name=“action" value=“REMOVE"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</DeleteRequest>

##### Example (Postpaid Activation)

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“Y"/>

<Parameter name=“bill\_cycle" value=“0101"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name="rfs" value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=”rfs” value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“CUG"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ member1" value=“9988453333"/>

..

<Parameter name=“ member5" value=“9988454444"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### Prepaid to Prepaid

Amdocs BIL sends request to deactivate the old prepaid service on the network elements and another request to activate a new different prepaid service.

##### Example (Prepaid Deactivation)

<DeleteRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>7</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“N"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name=“rfs” value=“MPVOICE"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“OCS\_OFFER"/>

<Parameter name=“action" value=“REMOVE"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</DeleteRequest>

##### Example (Prepaid Activation)

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“SMART Bro"/> -- proposed parameter to have sub-brand info like – SMART Bro Prepaid, SMART Bro Postpaid

<Parameter name=”mnp\_type” value=”INTRA/INTER”/> -- Proposed parameter for identifying between MNP (INTRA/INTER) and normal flow

<Parameter name=“is\_new\_account" value=“Y"/>

<Parameter name=“bill\_cycle" value=“0101"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<RFS>

<Parameter name="rfs" value="MPVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=”rfs” value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“CUG"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ member1" value=“9988453333"/>

..

<Parameter name=“ member5" value=“9988454444"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</CreateRequest>

### Inter MNP

#### Port Out (Postpaid)

Amdocs BIL sends request to deactivate the postpaid service on the network elements with mnp\_type=”INTER”.

##### Example (Postpaid Deactivation)

<DeleteRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>7</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“POS3G"/> -- proposed parameter to have sub-brand info like – ISM Brand code is expected

<Parameter name=”mnp\_type” value=”INTER”/> -- Proposed parameter for identifying between MNP and normal flow

<Parameter name=“is\_new\_account" value=“N"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<Parameter name=“operator" value=“AAA-H-BBB-CCCCDDEE-FGGGGGG"/> -- Newly added parameter, whereby AAA is from and BBB is to. (001=Globe, 002=SMART, 003=Dito).

<RFS>

<Parameter name=“rfs” value=“MVOICE"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“OCS\_OFFER"/>

<Parameter name=“action" value=“REMOVE"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</DeleteRequest>

#### Port Out (Prepaid)

Amdocs BIL sends request to deactivate the prepaid service on the network elements with mnp\_type=”INTER”.

##### Example (Prepaid Deactivation)

<DeleteRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>7</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>http://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“prepaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“POS3G"/> -- proposed parameter to have sub-brand info like – ISM Brand code is expected

<Parameter name=”mnp\_type” value=”INTER”/> -- Proposed parameter for identifying between MNP and normal flow

<Parameter name=“is\_new\_account" value=“N"/>

<Parameter name=“bill\_cycle" value=“01"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<Parameter name=“operator" value=“AAA-H-BBB-CCCCDDEE-FGGGGGG"/> -- Newly added parameter, whereby AAA is from and BBB is to. (001=Globe, 002=SMART, 003=Dito).

<RFS>

<Parameter name=“rfs” value=“MPVOICE"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMST"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“SMSO"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“ROAMING"/>

<Parameter name=“action" value=“REMOVE"/>

</RFS>

<RFS>

<Parameter name=“rfs” value=“OCS\_OFFER"/>

<Parameter name=“action" value=“REMOVE"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</DeleteRequest>

#### Port In (Postpaid Pre-Activation)

Amdocs BIL sends request to pre-activate the postpaid service on the network elements with mnp\_type=”INTER”.

##### Example (Postpaid Pre-Activation)

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“POS3G"/> -- proposed parameter to have sub-brand should be ISM Brand Code

<Parameter name=”mnp\_type” value=” INTER”/> -- Proposed parameter for identifying between MNP and normal flow

<Parameter name=“is\_new\_account" value=“Y"/>

<Parameter name=“bill\_cycle" value=“0101"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<Parameter name=“operator" value=“AAA-H-BBB-CCCCDDEE-FGGGGGG"/> -- Newly added parameter, whereby AAA is from and BBB is to. (001=Globe, 002=SMART, 003=Dito).

<RFS>

<Parameter name="rfs" value="MVOICE"/>

<Parameter name=“action" value=“PREACT"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### Port In (Postpaid Full Activation)

Amdocs BIL sends request to activate the postpaid service on the network elements with mnp\_type=”INTER”.

##### Example (Postpaid Activation)

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“POS3G"/> -- proposed parameter to have sub-brand should be ISM Brand Code

<Parameter name=”mnp\_type” value=” INTER”/> -- Proposed parameter for identifying between MNP and normal flow

<Parameter name=“is\_new\_account" value=“Y"/>

<Parameter name=“bill\_cycle" value=“0101"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<Parameter name=“operator" value=“AAA-H-BBB-CCCCDDEE-FGGGGGG"/> -- Newly added parameter, whereby AAA is from and BBB is to. (001=Globe, 002=SMART, 003=Dito).

<Parameter name=“so\_create\_date" value=“20191201160545"/>

<Parameter name=“sps\_so\_type" value=“xyz"/>

<Parameter name=“sne" value=“xyz"/>

<Parameter name=“tImsi" value=“xyz"/>

<Parameter name=“eki" value=“xyz"/>

<Parameter name=“icc\_id" value=“xyz"/>

<Parameter name=“algo\_id " value=“xyz"/>

<Parameter name=“kbd\_id" value=“xyz"/>

<Parameter name=“hlr" value=“xyz"/>

<Parameter name=“index\_code" value=“xyz"/>

<Parameter name=“pin1" value=“xyz"/>

<Parameter name=“pin2" value=“xyz"/>

<Parameter name=“puk1" value=“xyz"/>

<Parameter name=“puk2" value=“xyz"/>

<Parameter name=“sim\_type" value=“xyz"/>

<Parameter name=“sim\_capacity" value=“xyz"/>

<Parameter name=“sim\_status" value=“xyz"/>

<Parameter name=“welcome\_pack" value=“xyz"/>

<Parameter name=“device" value=“xyz"/>

<Parameter name=“lac" value=“xyz"/>

<Parameter name=“cell\_id" value=“xyz"/>

<Parameter name=“version" value=“xyz"/>

<Parameter name=“service\_name" value=“xyz"/>

<Parameter name=“user\_category" value=“xyz"/>

<Parameter name=“paid\_type" value=“xyz"/>

<Parameter name=“sub\_cos\_id" value=“xyz"/>

<RFS>

<Parameter name="rfs" value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=”rfs” value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“CUG"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ member1" value=“9988453333"/>

..

<Parameter name=“ member5" value=“9988454444"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</CreateRequest>

#### Change MSISDN (Postpaid)

Amdocs BIL sends request to change MSISDN.

##### Example (Change MSISDN)

<ModifyRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>1</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name="msisdn" value="MSISDN"/>

<Parameter name="imsi" value="IMSI"/>

<Parameter name="account\_number" value="ACCOUNT\_NUMBER"/>

<Parameter name="account\_type" value="ACCOUNT\_TYPE"/>

<Parameter name="bill\_cycle" value="BILL\_CYCLE"/>

<Parameter name="req\_date\_time" value="REQ\_DATE\_TIME"/>

<Parameter name="operator" value=“AAA-H-BBB-CCCCDDEE-FGGGGGG"/>

<RFS>

<Parameter name="rfs" value="**MVOICE**"/>

<Parameter name="action" value="**CHGNUMBER**"/>

<Parameter name="oldmsisdn" value="OLDMSISDN"/>

<Parameter name="udstype" value="UDSTYPE"/>

</RFS>

</RequestParameters>

</ModifyRequest>

#### Port In (Prepaid)

DSA sends request to L-USPS to activate the prepaid service on the network elements with example below.

##### Example (Prepaid Activation)

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:ins="http://soa.comptel.com/2011/02/instantlink">

<soap:Header/>

<soap:Body>

<ins:CreateRequest>

<ins:RequestHeader>

<ins:NeType>DSA</ins:NeType>

<ins:OrderNo>SmartPrepaid001</ins:OrderNo>

<ins:Priority>5</ins:Priority>

<ins:ReqUser>DSAUser1</ins:ReqUser>

</ins:RequestHeader>

<ins:RequestParameters>

<ins:Parameter name="SO\_CREATE\_DATE" value="20201030235132"/>

<ins:Parameter name="SNE" value="DSPRE"/>

<ins:Parameter name="SPS\_SO\_TYPE" value="MPRE"/>

<ins:Parameter name="MIN" value="639681329880"/>

<ins:Parameter name="IMSI" value="515301000000091"/>

<ins:Parameter name="T\_IMSI" value="515030103000565"/>

<ins:Parameter name="BRAND" value="SPP"/>

<ins:Parameter name="EKI" value="0730D6171A59C34BB6EEB8CD6FCBD08F"/>

<ins:Parameter name="ICCID" value="89630319123000000392"/>

<ins:Parameter name="ALGOID" value="000101"/>

<ins:Parameter name="KBDID" value="37"/>

<ins:Parameter name="HLR" value="HLRG1I"/>

<ins:Parameter name="INDEX\_CODE" value="45"/>

<ins:Parameter name="PIN1" value="1234"/>

<ins:Parameter name="PIN2" value="1234"/>

<ins:Parameter name="PUK1" value="03878162"/>

<ins:Parameter name="PUK2" value="34763205"/>

<ins:Parameter name="SIM\_TYPE" value="T92"/>

<ins:Parameter name="SIM\_CAPACITY" value="72"/>

<ins:Parameter name="SIM\_STATUS" value="A"/>

<ins:Parameter name="DEVICE" value="A"/>

<ins:Parameter name="LAC" value="A"/>

<ins:Parameter name="CELLID" value="A"/>

<ins:Parameter name="VERSION" value="LON02M5"/>

<ins:Parameter name="WELCOME\_PACK" value="WelcomeP1"/>

</ins:RequestParameters>

</ins:CreateRequest>

</soap:Body>

</soap:Envelope>

### Inter MNP (Non-DSA)

#### Port In (Postpaid Full Activation)

Amdocs BIL sends request to activate the postpaid service on the network elements with mnp\_type=”INTER”.

##### Example (Postpaid Activation)

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

<OrderNo>T001</OrderNo>

<Priority>2</Priority>

<ReqUser>OPTIMA</ReqUser>

<ReplyTo>https://optesbdev2:8085/v1/AccountCustomManagement/handleResponse</ReplyTo>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_number" value=“897454"/>

<Parameter name=“account\_type" value=“postpaid"/> -- Existing parameter, value need to be changed based on service type

<Parameter name=“brand" value=“SMART"/> -- proposed parameter to have brand info – SMART/SUN

<Parameter name=“sub\_brand" value=“POS3G"/> -- proposed parameter to have sub-brand should be ISM Brand Code

<Parameter name=”mnp\_type” value=” INTER”/> -- Proposed parameter for identifying between MNP and normal flow

<Parameter name=“is\_new\_account" value=“Y"/>

<Parameter name=“bill\_cycle" value=“0101"/>

<Parameter name=“req\_date\_time" value=“20191201160545"/>

<Parameter name=“operator" value=“AAA-H-BBB-CCCCDDEE-FGGGGGG"/> -- Newly added parameter, whereby AAA is from and BBB is to. (001=Globe, 002=SMART, 003=Dito).

<Parameter name=“so\_create\_date" value=“20191201160545"/>

<Parameter name=“sps\_so\_type" value=“xyz"/>

<Parameter name=“sne" value=“xyz"/>

<Parameter name=“tImsi" value=“xyz"/>

<Parameter name=“eki" value=“xyz"/>

<Parameter name=“iccid" value=“xyz"/>

<Parameter name=“algo\_id " value=“xyz"/>

<Parameter name=“kbd\_id" value=“xyz"/>

<Parameter name=“hlr" value=“xyz"/>

<Parameter name=“index\_code" value=“xyz"/>

<Parameter name=“pin1" value=“xyz"/>

<Parameter name=“pin2" value=“xyz"/>

<Parameter name=“puk1" value=“xyz"/>

<Parameter name=“puk2" value=“xyz"/>

<Parameter name=“sim\_type" value=“Non-DSA"/> -- Either DSA/Non-DSA

<Parameter name=“sim\_capacity" value=“xyz"/>

<Parameter name=“sim\_status" value=“xyz"/>

<Parameter name=“welcome\_pack" value=“xyz"/>

<Parameter name=“device" value=“xyz"/>

<Parameter name=“lac" value=“xyz"/>

<Parameter name=“cell\_id" value=“xyz"/>

<Parameter name=“version" value=“xyz"/>

<Parameter name=“service\_name" value=“xyz"/>

<Parameter name=“user\_category" value=“xyz"/>

<Parameter name=“paid\_type" value=“xyz"/>

<Parameter name=“sub\_cos\_id" value=“xyz"/>

<RFS>

<Parameter name="rfs" value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=”rfs” value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“CUG"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ member1" value=“9988453333"/>

..

<Parameter name=“ member5" value=“9988454444"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“ROAMING"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“OCS\_OFFER"/>

<Parameter name=“action" value=“ADD"/>

<Parameter name=“ocs\_offer\_id" value=“77838383"/>

</RFS>

</RequestParameters>

</CreateRequest>

## CDPA - High Level Workflow

The following explains the high-level logic flow of multiple line item/multiple service order processing for Nokia CDPA solution:



Figure 8. Multiple line items processing flow



Figure 9. Single line item processing flow

Following table describes different logic component which allows extension:

Table 7. Generic BST framework extension

| **Logic** | **Description** |
| --- | --- |
| Initialization | * Execute one time per request * BST constant parameter definition * E.g.: NE\_ID definition, load balancing logic for deriving NE\_ID, etc. |
| Pre-Service handling | * Execute multiple time per request, once for each line item * Message adaptation, internal parameter initialization * Logic library invocation * Lookup table invocation |
| Pre-Execution handling | * Control by reserved task parameter defined in Catalog * Special handling before sending to Network. * Customized logic libraries invocation to support special case handling on specific products. * Generic logic libraries to perform common function used in the main business flow. |
| Post- Execution handling | * Control by reserved task parameter defined in Catalog * Special Handling after sending to network * Customized logic libraries to support special case handling on specific products. * Generic logic libraries to process logic rules defined in technical services. |

### Non CDPA – Optima Notification

In legacy USPS, there is also an enhancement required on existing BST to support sending provisioning status (form of notification) of non-Optima initiated northbound requests to Optima. Below is the list of non-optima north bound systems, the command to call and the use cases supported.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **North Bound System** | **North Bound API** | **Command** | **Use Cases** | **Notes** |
| DSA | Online SOAP | Optima SimSwap  (REST) | DSA SimSwap Smart and Sun Prepaid/Postpaid |  |
| VASPROV | Online SOAP | UpdateAddOnRegistration API (REST) | Feature Activation and Feature Deactivation | A prodtype-sotype-sne to offerid mapping is maintained |
| OneFMS | Online SOAP | Optima File/Ftp | Subscriber Tagging |  |

#### Subscriber Tagging

| File Format : | BatchFMS\_20170814000051.req |
| --- | --- |
| Content: | NE\_TYPE ORDER,NE\_ID hlr1,REQ\_TYPE 1,REQ\_USER Optima,ORDER\_TYPE Modify,IMSI1 99999988453333,MSISDN1 9988453333,ACCOUNT\_NUMBER 435455,ACCOUNT\_TYPE Postpaid, BILL\_CYCLE P01,REQ\_DATE\_TIME 20191201160545,SO1\_TECHNICALPRODUCT MVOICE,SO1\_ACTIONCODE ADD,SO1\_SERVICE\_UNIQUE\_ID 026462771,SO1\_SERVICE\_TYPE SMART,SO1\_ACCOUNT\_ACTION Add\_FBL,SO1\_ACCOUNT\_REASON Fraudulent Account,SO1\_SERVICE\_ACTION BAR\_BIN,SO1\_SERVICE\_REASON Due to Fraud,SO1\_BL\_DATA\_ACTION Add,SO1\_BL\_DATA\_REASON Address found as fraudulent,SO1\_LOT\_HOUSE\_NUMBER 37,SO1\_ROOM\_FLOOR 3,SO1\_BUILDING\_APARTMENT\_NAME SMART Tower,SO1\_STREET\_NAME Ayala Avenue,SO1\_SUBDIVISION\_VILLAGE ,SO1\_BARANGAY ,SO1\_CITY Makati,SO1\_PROVINCE Manila,SO1\_TITLE Mr,SO1\_GIVEN\_NAME John,SO1\_MIDDLE\_NAME Tiberius,SO1\_SURNAME Smith,SO1\_SUFFIX ,SO1\_BIRTHDATE 1989-07-16,SO1\_COUNTRY\_CODE 0063,SO1\_AREA\_CODE 97,SO1\_TELEPHONE\_NUMBER 87654321,SO1\_LOCAL\_NUMBER 12345678,SO1\_EMP\_BUSINESS\_NAME PLDT,SO1\_EMP\_LOT\_HOUSE\_NUMBER 37,SO1\_EMP\_ROOM\_FLOOR 3,SO1\_EMP\_BUILDING\_APARTMENT\_NAME SMART Tower,SO1\_EMP\_STREET\_NAME Ayala Avenue,SO1\_EMP\_SUBDIVISION\_VILLAGE ,SO1\_EMP\_BARANGAY ,SO1\_EMP\_CITY Makati,SO1\_EMP\_PROVINCE Manila,SO1\_EMP\_BUSINESS\_CONTACT Jane Smith,SO1\_ID\_CARD\_TYPE Personal ID,SO1\_ID\_NUMBER 12345678,SO1\_ENTRY\_ID 555555,SO1\_USER\_ID 12345,SO1\_USER\_ROLE 457,SO1\_ENTRY\_DATE 20190129150507 |

## Special Handling

This section will discuss any special handling that will be implemented in this Phase.

### Critical NE handling

A logic will be implemented to prioritize Critical NEs and check for the provisioning result status from each NEs and trigger the rollback (for eligible scenarios) if any of the critical NE fails. The same logic will handle sending an intermediate notification to Optima for the status of Critical NE provisioning.

### RFS Sorting

RFS sorting is required to cater same RFS name but different Actions. The logic will trigger Remove first, then Add.

The sequencing on HLR commands will only be based on the batch request spml format and dependency criteria to be provided by Amdocs.

### RFS Aggregation

FlowOne is required to decompose multiple HLR related RFSs received from Optima and provision only single call to HLR NE i.e. multiple RFSs – single provisioning call to the target NE behaviour. A “Pre-Processing” library/hook between Initialization and the LineItemProcessing that will decompose all line items and aggregate the tasks before executing the command will be implement. The following requirement is included in the logic.

* TS11 RFS needs to always be included in the first sequence of batchRequest provisioning to the HLR during Create scenario

The final HLR batch request format is a generic spml template to be provided by Amdocs.

### CRI Inquiry

For All wireless transactions, FlowOne needs to inquire from CRI to determine if MSISDN is migrated and which HLR campus it belongs, based on value returned for GT\_ADDRESS and STATUS. The HLR campus will be mapped to the corresponding spml command to provision to HLR.

Table 8. CRI Inquire Value Mapping

|  |  |  |
| --- | --- | --- |
| **GT\_ADDRESS|STATUS** | **HLR\_CAMPUS** | **Special Handling** |
| 0|0 | Legacy | If MSISDN is still in legacy, Fail the SO and will return error to Optima. Provisioning will terminate. |
| 22|639180001841 | IDC | Provision HLR via IDC spml commands defined in Trident command mapping sheet |
| 22|639180001842 | CEBU | Provision HLR via IDC spml commands defined in Trident command mapping sheet |

Note: If MSISDN is legacy, the SO will be FAILED and provisioning will be terminated.

### ISM Inquiry

This is applicable to RFS=MVOICE and FLP, ACTION is REMOVE. Since ICCID is not provided in SO request parameters, FlowOne will inquire from ISM to get the needed ICCID for DP9 permanent disconnect.

For MVOICE\_ADD, ISM inquiry will also be triggered to get ICCID value for CWS provisioning.

### DP9 Permanent Disconnect Special Handling

The handling requires calling display command from DP9 to get the valid ICCID value that will be used for disconnect, in cases disconnect via the ICCID retrieved from ISM fails. The newly retrieved ICCID will then be used to purge the sim from DP9. This fallout handling will be implemented for RFS MVOICE and FLP, ACTION is REMOVE.

### Intra MNP

Intra MNP request will be determined by checking the value of the new parameter “mnp\_type” in the request. If the parameter exists, this will be either for Intra MNP or Inter MNP. If the value is “INTRA” then the request is for Intra MNP.

### Inter MNP

Inter MNP request will be determined by checking the value of the new parameter “mnp\_type” in the request. If the parameter exists, this will be either for Intra MNP or Inter MNP. If the value is “INTER” then the request is for Inter MNP. Additionally for SIM\_TYPE, checking will be done if SIM\_TYPE=Non-DSA or not to determine the provisioning flow. This handling is only specifically for MVOICE RFS.

## Response Handling

There are two types of response to be sent from FlowOne to Optima, as below:

1. Notification – this is an intermediate response to notify Optima the status of provisioning for Critical NEs. It is defined as ResponseType = 1 in the command parameter mapping.
2. Final Response – this is the final response notifying Optima of the overall status of provisioning to all NEs (including NonCritical NEs). It is defined as ResponseType=2 in the command parameter mapping.

### Intermediate Notification via handleResponse (command and response format)

**Success Case:**

{

"ResponseHeader": [

                          {

                                                "RequestId": "4514315",

                                                "Status": "9",

                                                "OrderNo": "71632282",

                                                "StatusMessage": "Request Execute Successfully",

                                                "StatusMessageId": "000",

                                                "Priority": "5",

                                                "ReqUser": "OPTIMA",

                                                "ReceivedDate": "2018-05-02T14:48:03.947+08:00",

                                                "FinishedDate": "2018-05-02T15:24:06.773+08:00"

                        }

                        ],

                        "ResponseParameters": [

                          {

                                                  "ResponseType" : "1",

                                                  "NeType":"ORDER",

                          }

                        ]

}

**Failed Case:**

{

                        "ResponseHeader": [

                          {

                                                "RequestId": "4514315",

                                                "Status": "7",

                                                "OrderNo": "71632282",

                                                "StatusMessage": "ERROR MESSAGE",

                                                "StatusMessageId": "BST11",

                                                "Priority": "5",

                                                "ReqUser": "OPTIMA",

                                                "ReceivedDate": "2018-05-02T14:48:03.947+08:00",

                                                "FinishedDate": "2018-05-02T15:24:06.773+08:00"

                        }

                        ],

                        "ResponseParameters": [

                          {

                                                  "ResponseType" : "1",

                                                  "NeType":"ORDER",

                                                  "NEId1 ":"pgw1",

                                                  "RFS\_CODE1":"MVOICE",

                                                  "ERROR\_CODE1":"130",

                                                  "ERROR\_MSG1":"Subscriber address invalid"                                      ....

                          }

                        ]

}

### Final Response via handleResponse (command and response format)

**Success Case:**

{

"ResponseHeader": [

                          {

                                                "RequestId": "4514315",

                                                "Status": "9",

                                                "OrderNo": "71632282",

                                                "StatusMessage": "Request Execute Successfully",

                                                "StatusMessageId": "000",

                                                "Priority": "5",

                                                "ReqUser": "OPTIMA",

                                                "ReceivedDate": "2018-05-02T14:48:03.947+08:00",

                                                "FinishedDate": "2018-05-02T15:24:06.773+08:00"

                        }

                        ],

                        "ResponseParameters": [

                          {

                                                  "ResponseType" : "2",

                                                  "NeType":"ORDER",

                          }

                        ]

}

**Failed Case:**

{

                        "ResponseHeader": [

                          {

                                                "RequestId": "4514315",

                                                "Status": "7",

                                                "OrderNo": "71632282",

                                                "StatusMessage": "ERROR MESSAGE",

                                                "StatusMessageId": "BST111",

                                                "Priority": "5",

                                                "ReqUser": "OPTIMA",

                                                "ReceivedDate": "2018-05-02T14:48:03.947+08:00",

                                                "FinishedDate": "2018-05-02T15:24:06.773+08:00"

                        }

                        ],

                        "ResponseParameters": [

                          {

                                                  "ResponseType" : "2",

                                                  "NeType":"ORDER",

                                                  "NEId1 ":"cws",

                                                  "RFS\_CODE1":"MVOICE",

"ERROR\_CODE1":"130",

                                                  "ERROR\_MSG1":"Subscriber address invalid",

"NEId2 ":"uvc",

"RFS\_CODE2":"MVOICE",

                                                  "ERROR\_CODE2":"130",

                                                  "ERROR\_MSG2":"Subscriber address invalid",

                                                  ....

                          }

                        ]

}

### HandleResponse command response format

**Success Case:**

{

“code”:0,

“message”:”Success”,

“orderId”:”88888”

}

**Failed Case:**

{

“code”:55,

“message”:”Error Message”,

“orderId”:”88888”

}

**Note**: If response sending fails, a retry will be executed once after a given interval. It is not recommended to handle multiple retries as that will impact the application performance. Number of retry and interval can be configured via lookup table.

## Logic Libraries

This section lists the Logic Libraries needed in the current phase. They will be reused across multiple project phases and new logic libraries may also be defined in next phases.

Table 9. Logic Libraries

|  |  |
| --- | --- |
| Logic Library | Description |
| Error Whitelisting | Logic library to implement BST level whitelisting. |

### Error/Rollback Handling

### Error Whitelisting

The logic should have capabilities to whitelist an error on BST or NE level based on criteria that form a generic identifier for a task.

Error search key (BST)

* NAME
* REQ\_TYPE (optional)
* ACTION (optional)
* NE\_ERROR\_CODE
* REPUSH (TRUE/FALSE)

BST level whitelisting allows an additional parameter REPUSH to be included in the request to trigger on BST level whitelisting. The parameter will only be available for repush requests/scenarios. All regular requests including first time provisioning won’t require the parameter. Given this, whitelisting is only enabled during repush and is disabled unless REPUSH parameter is provided and set to TRUE. BST/Catalog can define the logic based on the existence of the parameter and the list of error codes for whitelisting.

Sample request for repush:

<CreateRequest>

<RequestHeader>

<NeType>ORDER</NeType>

 <OrderNo>T001</OrderNo>

<Priority>3</Priority>

<ReqUser>Optima</ReqUser>

</RequestHeader>

<RequestParameters>

<Parameter name=“msisdn" value=“9988453333"/>

<Parameter name=“imsi" value=“99999988453333"/>

<Parameter name=“account\_no" value=“897454"/>

<Parameter name=“account\_type" value=“Postpaid"/>

<Parameter name=“bill\_cycle " value=“M01"/>

<Parameter name=“repush" value=“TRUE"/>

<Parameter name=“req\_date\_time " value=“20191201160545"/>

<RFS>

<Parameter name="“rfs" value="MVOICE"/>

<Parameter name=“action" value=“ADD"/>

</RFS>

<RFS>

<Parameter name=“rfs" value=“SMST"/>

<Parameter name=“action" value=“ADD"/>

</RequestParameters>

</CreateRequest>

Note: REPUSH parameter handling is not applicable to NE level whitelisting since the status lookup file format is standard to all NEIs.

NEI Whitelisting status\_lookup file content:

<ERROR CODE>,<ACTION>,<CONN STATUS>,<TASK STATUS>,<STATUS MESSAGE>

### Rollback Handling

Rollback handling will be implemented for eligible scenarios as follows:

* + - When provisioning request is received from Amdocs Ordering via BIL, USPS will decompose the requests into Critical and Non-Critical.
    - Critical NEs are processed before processing Non-Critical NEs.
    - If one of the Critical NEs in a transaction fails, then USPS will try to roll back the whole transaction including all the previously successful Critical NEs and will send failure response to Amdocs Ordering.
    - No rollback required for the Non Critical NEs.

Rollback Eligibility Conditions:

* USPS will try to rollback Critical RFs for all use cases if there is a corresponding rollback command for RFS. If there is no rollback defined for RFS, then no handling for that RFS.
* Rollback commands will be provided by customer. If there is no rollback provided for RFS, then no rollback will be implemented.
* Provisioning Network Element Errors will be handled manually by Amdocs Operations (IRM) using Amdocs SPS GUI for those failed RFS without rollback commands.
* Rollback of provisioning between FlowOne to HLR via PGW shall follow existing behavior where PGW is capable to handle the rollback in HLR for the failed scenario.

## Catalog HLD

### Catalog Item

The section will be describing the items that are going to be used in the Catalog Driven Provisioning and Activation solution implementation.

Catalog items will consist of:

* **Resource Facing Service (RFS)** is a feature offered by network. It is an “object” that can be activated and deactivated (or created and deleted).
* **Technical Services** **(TS)** are the commends provided by network element. They are represented by action/tasks supported NEI.
* **Selection Group** **(SG)** is a container for a group of items triggered by one or more conditions
* **Technical Libraries** **(TL)** are remote Items that, instead of representing a single task to be executed on the Network, they represent a Logic Library to be executed. They can be a collection of Technical Services.

### Resource Facing Services Actions

RFS Action is the “Provisioning scenario” driven by the type of message (or command) received from BSS system or other Northbound system, so it maps into a corresponding transaction in the catalog, with reference to Request Type.

Below are the defined RFS Action for Optima provisioning:

* ADD
* REMOVE
* SUSPEND
* RESUME
* CHGNUMBER
* CHGIMSI
* CHGBILLCYCLE
* CHGOWNERSHIP
* ND
* NDLIFT

### Request Types

Request Type is the type of request received from BSS system or other Northbound system which is modelled together with RFS Action to determine the specific transaction to invoke from the Catalog.

As a set of generic Request Types supported for Trident Optima could be:

* CreateRequest
* ModifyRequest
* DeleteRequest

### Catalog Modelling

### Sample RFS Model

This section represents sample RFS model for Add Service Transaction. The model include the RFS and the TS embedded to it.

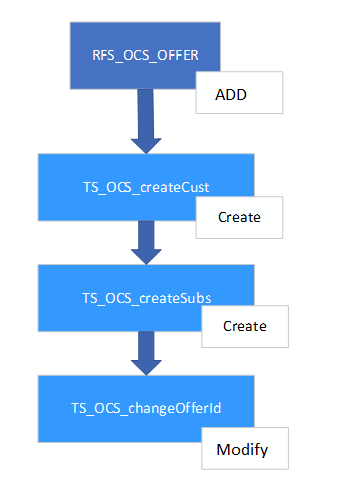


Figure 10. Sample RFS Model

### RFS

Selected RFS mentioned in this section will be delivered as part of Phase 1a. Same design principles apply for other RFSs and future phases. The complete list can be found in next section.

#### DATA

RFS for HLR Data Service. Decomposes into TS directly



#### OCS OFFER

RFS for OCS Offer/Add Ons. Decomposes into TS directly



#### MVOICE

RFS for Voice Basic Service. Decomposes into TS directly



### Services to RFS mapping

Below are the Services and supported RFS(s). Nokia will only receive RFS list corresponding to specific service that are only visible to the northbound. Nokia will not model Services in catalog.

Table 10. Phase 1a - Service to RFS Mapping

|  |  |
| --- | --- |
| Service | RFS |
| Basic Mobile | MVOICE  LTE  SMSO  SMST  DATA  CALID  CFU  CFNRY  CFNRC  CFB |
| Mobile Plan Add Ons | OCS\_OFFER  IDD |

Table 11. Phase 1b - Service to RFS Mapping

|  |  |
| --- | --- |
| Service | RFS |
| Basic Mobile | CALLHOLD  MULTIPARTY |
| Mobile Plan Add Ons | BAROUTGOING  ROAMING  DATAROATMING |
| Mobile Other Add Ons | BARINCOMING  VOLTE |
| FLP | FLP |
| IPTV | IPTVCUSTOMER  IPTVPPV |
| SAT TV | SATTV  SATTVPPV SATTVTP |
| Notice of Disconnect | VOIP |

Table 12. Intra MNP – Service to RFS Mapping

|  |  |
| --- | --- |
| Service | RFS |
| Intra MNP | MVOICE  MPVOICE |

Table 13. Inter MNP – Service to RFS Mapping

|  |  |
| --- | --- |
| Service | RFS |
| Inter MNP | MVOICE  MPVOICE |

Table 14. Fixed Wireless – Additional RFS

|  |  |
| --- | --- |
| Service | RFS |
| Fixed Wireless | ACLOC  ACNDD  ACOAC  ACSAC  ACSRV |

### Undo Transactions

In case a transaction fails during processing, Catalog attempts to undo any transactions on the embedded items that have already completed. This is possible by configuring another transaction as the undo transaction. During rollback management, Logic Library can fetch the undo transaction decomposition and execute using NEI’s.

In the below example, MVOICE RFS, ‘DELETE’ transaction is defined as UNDO transaction for the ‘CREATE’ transaction.



Figure 11. Undo Transaction

# Non-Functional Requirement

This section is covered in another HLD (PLDT FlowOne19 Installation High Level Design).

# Appendix A (RFS Catalog)

## RFS Catalog

|  |  |
| --- | --- |
| **Columnname** | **Description** |
| MVOICE | Voice/Intra MNP/Inter MNP (Postpaid) |
| CALID | Caller Id |
| CALLHOLD | Call Holding |
| MULTIPARTY | Multiparty (MPTY) Call |
| SMSO | Short Message Outgoing |
| SMST | Short Message Terminating |
| CFU | Call Forwarding Unconditional |
| CFNRY | Call Forwarding on No Reply |
| CFNRC | Call Forwarding on Mobile Subscriber Not Reachable |
| CFB | Call Forwarding on Mobile Susciber Busy |
| DATA | Data |
| LTE | Smart LTE |
| CUG | Called User Group |
| OCS\_OFFER | Mobile Plan Add-Ons |
| BAROUTGOING | Barring of Outgoing Voice |
| ROAMING | INTERNATIONAL ROAMING |
| IDD | IDD |
| BARINCOMING | Subscriber Determined Barring Incoming |
| VOLTE | Volte |
| DATAROAMING | Data Roaming |
| FLP | FLP Replenishment |
| IPTV | IPTV |
| SATTV | SATTV |
| VOIP | For Notice of Disconnect |
| MPVOICE | For Intra/Inter MNP (Prepaid) |
| ACLOC | LOCAL (for fixed wireless) |
| ACNDD | NDD (for fixed wireless) |
| ACOAC | OPERATOR ASSISTED CALL (for fixed wireless) |
| ACSAC | DDD SECURITY ACCESS CODES (for fixed wireless) |
| ACSRV | ACCESS ON SERVICE NUMBERS (for fixed wireless) |

# Appendix B (Reference Documents)

The following documents were used as references for this HLD:

|  |  |
| --- | --- |
| No. | Document Name |
| 1 | MEC OPTIMA SOM USPS RFS Mapping V19.xlsx |
| 2 | Trident Wireless Optima RFS Command Mapping\_202006101002.xlsx |
| 3 | PLDT - Trident - 2b - Optima Postpaid USPS IDD\_14.docx |
| 4 | USPS NEI List\_ updated v911082019.xlsx |
| 5 | PLDT - Trident - 2b - Optima-USPS(MNP) IDD.docx |
| 6 | PLDT - Inter MNP-Optima-USPS IDD v2.0.docx |

# Appendix C (Document Change Tracker)

The following tracks the changes in requirements reflected in this HLD:

|  |  |  |
| --- | --- | --- |
| No. | Description of Change | Date Confirmed |
| 1 | WSO2 Request Format payload from SOAP to XML (sec. 4.4.1-4.4.2) | 10-Feb-2020 |
| 2 | HandleResponse from SOAP to REST Json (sec 4.7) | 19-Feb-2020 |
| 3 | Phase1B requirement Updates (sec 4.6.5-4.6.7) | 14-Feb-2020 |
| 4 | RFS List supported for Phase1a and Phase1b was redefined (4.10.8 and 6.1) | 12-Mar-2020 |
| 5 | mWARE removed from scope for Optima Notification due to decommissioning (4.5.1) | 13-Mar-2020 |

# Appendix D (Change Tracker)

The following list of changes are reflected as additional requirements on top of Trident Optima-USPS solution.

|  |  |  |
| --- | --- | --- |
| No. | Description | Date Added |
| 1 | SUN FLP Replenishment via Batch file | 02-09-2020 |
| 2 | ARM inquire for Notice of Disconnect and Change Bill Cycle Wireline | 02-09-2020 |
| 3 | Fixed Wireless and Home Wifi | 02-09-2020 |
| 4 | IRAFM header and content format | 02-09-2020 |

# HLD Acceptance

Upon signing below, both parties accept this Optima HLD in its current form.

|  |  |  |
| --- | --- | --- |
| **Prepared by Nokia:**  **Signed**:  **Name**: Kamhoong Lai  **Title**: Domain Architect  **Date**:08-12-2021 | **Reviewed by Nokia:**  **Signed**:  **Name**: Parin Wongchamboon  **Title**: Project Manager  **Date**:08-12-2021 | **Approved by Nokia:**  **Signed**:    **Name**: Parin Wongchamboon  **Title**: Project Manager  **Date**:08-12-2021 |

|  |  |  |
| --- | --- | --- |
| **Approved by Amdocs:**  **Signed**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_    **Name**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_    **Title**:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_    **Date**:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | **Approved by Amdocs:**  **Signed**:    **Name**:    **Title**:    **Date**: | **Approved by Amdocs:**  **Signed**:    **Name**:    **Title**:    **Date**: |