Oracle Utilities Notifications Hub

Also applicable to:
Oracle Utilities Customer Cloud Service
Oracle Utilities Customer to Meter
Oracle Utilities Customer Care and Billing Cloud Service

Configuration Guide
Release 24A
F99486-01

June 2024

ORACLE®

Oracle Utilities Notifications Hub Configuration Guide

Contents

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

Chapter 7

Chapter 8

Chapter 9

Chapter 10

Preface

Welcome to the Oracle Utilities Notifications Hub Configuration Guide for release 24A

The preface includes the following:

- Audience
- Documentation and Resources
- Updates to Documentation
- Documentation Accessibility
- Conventions
- Acronyms

Audience

This document is intended for anyone implementing the Oracle Utilities Notifications Hub integration.

Documentation and Resources

For more information regarding this integration, foundation technology and the edge applications, refer to the following documents:

Product Documentation

Topic	Location
Oracle Utilities Notifications Hub integration documentation	https://docs.oracle.com/en/industries/energy-water/cloud-integrations/index.html
Oracle Utilities Customer Cloud Service documentation	https://docs.oracle.com/en/industries/energy-water/customer-cloud-service/index.html
Oracle Utilities Customer Care and Billing documentation	https://docs.oracle.com/en/industries/energy-water/ccb/index.html
Oracle Utilities Customer to Meter documentation	https://docs.oracle.com/en/industries/energy-water/advanced-meter/index.html
Oracle Utilities Network Management System documentation	https://docs.oracle.com/en/industries/energy-water/network-management-system/index.html
Oracle Responsys documentation	https://docs.oracle.com/en/cloud/saas/marketing/responsys.html

Additional Documentation

Resource	Location
Oracle Integration Cloud Service documentation	Refer to the OIC documentation at: https://docs.oracle.com/en/cloud/paas/integration-cloud/index.html
Oracle Support	Visit My Oracle Support at https://support.oracle.com regularly to stay informed about updates and patches. Refer to the *Certification Matrix for Oracle Utilities Products (Doc ID 1454143.1)* on My Oracle Support to determine if support for newer versions of the listed products is included. For more information, refer to the Oracle Utilities Integrations page at http://my.oracle.com/site/tugbu/productsindustry/productinfo/utilities/integration/index.htm
Oracle University for training opportunities	http://education.oracle.com/

Updates to Documentation

The complete Oracle Utilities Notifications Hub documentation set is available from Oracle Help Center at https://docs.oracle.com/en/industries/energy-water/index.html.

Visit My Oracle Support for additional and updated information about the product.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support for the hearing impaired. Visit: http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Acronyms

The following terms are used in this document:

Term	Expanded Form
OUCCB/CCB	Oracle Utilities Customer Care and Billing
OUC2M/C2M	Oracle Utilities Customer to Meter
OUNMS/NMS	Oracle Utilities Network Management System
OIC	Oracle Integration Cloud
DVM	Domain Value Map (Lookup)
Responsys	Oracle Responsys
OCI Stream	Oracle Cloud Infrastructure Stream

Term	Expanded Form
OU	Oracle Utilities

Chapter 1

Introduction

This chapter provides an overview about the integration between Oracle Utilities Customer Cloud Service and Oracle Utilities Network Management System using Oracle Integration Cloud. It focuses on software requirements, Oracle Integration Cloud and business standpoint of the integration.

The chapter includes the following:

- Overview of the Integration
- About Oracle Utilities Customer Cloud Service
- About Oracle Utilities Customer to Meter
- About Oracle Utilities Customer Care Billing
- About Oracle Utilities Network Management System
- About Oracle Responsys
- About Oracle Integration Cloud
- About Oracle Cloud Infrastructure
- Software Requirements
- Prerequisites

Important! This integration is also applicable to the following:

- Oracle Utilities Customer Cloud Service (CCS)
- Oracle Utilities Customer to Meter (C2M)
- Oracle Utilities Customer Care and Billing Cloud Service (CCBCS)

Please note that this integration applies to any Oracle product that includes Oracle Utilities Customer Care and Billing.

Overview of the Integration

The Oracle Utilities Notifications Hub integration enables outbound, Oracle Utilities Customer Cloud Service-related and Oracle Utilities Network Management System (Outage)-related messages to be sent to customers via Oracle Responsys. The primary systems involved are Oracle Utilities Network Management System (NMS), Oracle Utilities Customer Cloud Service (CCS), Oracle Integration Cloud (OIC), and Oracle Responsys. In addition, Oracle Utilities Notifications Hub can be leveraged by other on-premises and cloud applications to send domain-specific messaging.

For any outage-related events (such as Outage Scheduled/Started, ERT updates, etc.) created in Oracle Utilities Network Management System, notification messages will be created for all accounts affected by a specific outage. Next, the integration will leverage customer account channel preferences from Oracle Utilities Customer Cloud Service to enrich and assemble the outage notification message content based on the desired target channels. The messages will then be sent to Oracle Responsys, where based on a previously configured e-mail and/or SMS campaigns, outage communications will be distributed to affected customers.

A separate synchronization process is used to update Oracle Utilities Network Management System's notification preferences (account-level), based on the account notification preferences specified in Oracle Utilities Customer Cloud Service.

For any Oracle Utilities Customer Cloud Service-related events (Bill Due/Past Due, Collection/Disconnect Warning, etc.), the integration will fetch outbound notification messages from Oracle Utilities Customer Cloud Service queues and send the complete messages to Oracle Responsys for distribution to customers' preferred channels, based on pre-configured email and/or SMS Campaigns.

To enable customer opt-outs for email notifications without logging into a self-service portal or contacting the call center to modify their enrollment preference, this integration supports unsubscribe links for transactional emails. Additionally, version 22B includes enhancements to Customer Contacts to process event data, such as Sent/Failed from Oracle Responsys and update the status on the Customer Contact record in Oracle Utilities Customer Cloud Service.

This integration can be leveraged to:

- Send outage notifications/updates to customers across multiple channels.

- Send customer-related notifications to customers across multiple channels.

- Obtain customer notification and delivery preferences from Oracle Utilities Customer to Meter (notifications preferences/types, delivery channels).

- Utilize campaign management in Oracle Responsys, to create and customize e-mail/SMS campaigns based on utilities' needs.

- Create a Customer Contact log for both inbound and outbound contacts.

- Support Customer Unsubscribe(s) for Email Preferences.
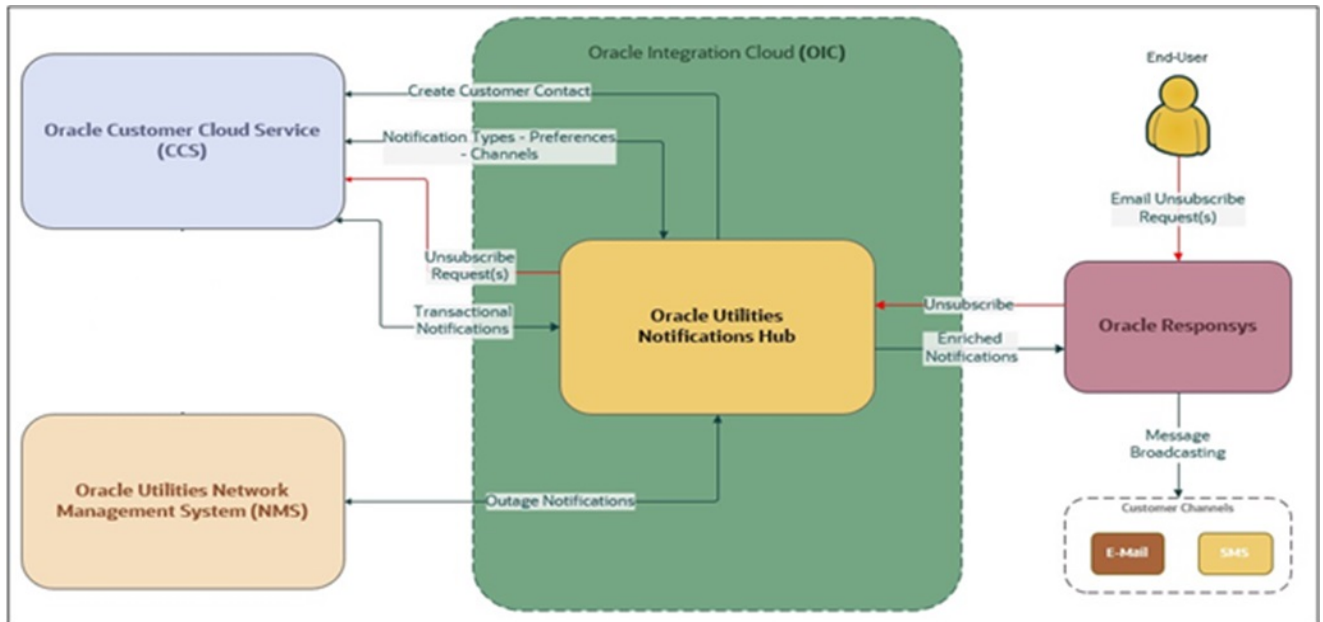
About Oracle Utilities Customer Cloud Service

Oracle Utilities Customer Cloud Service is a customer care, service order, metering and billing solution. The solution is designed to cater for utilities of all sizes, supports one to many utilities' service types, and handles the complexities associated with a utility's processes.

About Oracle Utilities Customer to Meter

Oracle Utilities Customer to Meter (C2M) brings together market-leading customer care and metering in a one solution, running on one database, accessed through one screen-to deliver connected operations and more valuable connections with your customers.

About Oracle Utilities Customer Care Billing

Oracle Utilities Customer Care and Billing is a central repository for customer information, such as name, address, phone number, and so on which manages all aspects of the utility customer lifecycle.

Oracle Utilities Customer Care and Billing also supports sending notifications and the maintenance of communication preference for notification types owned by Oracle Utilities Customer Care and Billing and other edge applications, such as Oracle Utilities Network Management System.

About Oracle Utilities Network Management System

Oracle Utilities Network Management System processes trouble calls from customers and analyzes them to determine probable outage locations. It can generate estimated restoration times (ERTs) that can then be provided back to customers.

Oracle Utilities Network Management System also keeps a history of all of the customer calls that were entered in the system, as well as a history of all events that were known to affect a customer even if the customer did not call in. In addition to responding to unplanned outages and non-outage problems, Oracle Utilities Network Management System can help a utility plan maintenance work or new construction that may impact existing customers. When the detailed switching plans are generated in Oracle Utilities Network Management System, information can be provided to customers about planned outages that will impact them.

About Oracle Responsys

Oracle Responsys Campaign Management is a cross-channel campaign management platform that delivers advanced intelligence at scale so you can create personalized messages based on the individual interests and preferences of customers and prospects.

About Oracle Integration Cloud

Oracle Integration Cloud is a unified platform to integrate the applications, automate processes, and create applications.

Using the Process Builder, the business processes can be rapidly designed, automated, and managed in the cloud. Using integrations connect the applications into a continuous business flow. The applications that are on cloud and on premises can be integrated using Oracle Integration Cloud. The lookups help to match application specific codes between the two applications.

Integration Insights and Stream Analytics helps to simplify and extract business metrics and create custom dashboards.

About Oracle Cloud Infrastructure

Oracle Cloud Infrastructure is a set of complementary cloud services that enable you to build and run a wide range of applications and services in a highly available hosted environment. Oracle Cloud Infrastructure (OCI) offers high-performance compute capabilities (as physical hardware instances) and storage capacity in a flexible overlay virtual network that is securely accessible from your on-premises network.

Software Requirements

The following software is required for the integration to work:

- Oracle Utilities Customer to Meter

- Oracle Utilities Customer Cloud Service

- Oracle Integration Cloud

- Oracle Utilities Network Management System

- Oracle Responsys

- OCI Streaming

For specific application versions, refer to the *Oracle Utilities Notifications Hub Release Notes* included in this release.

Prerequisites

The following are the prerequisites for this integration:

- This integration requires Oracle Utilities Network Management System to be V2.6.0.1.3 at minimum.

  Refer to the *Oracle Utilities Customer to Meter Integration to Oracle Utilities Network Management System Implementation Guide* for more information. It is available at: https://docs.oracle.com/en/industries/energy-water/integrations-index.html

- This integration requires two Oracle Integration Cloud Streams named SourceNotificationStream and TargetNotificationStream with at least one partition.

  Refer to https://docs.oracle.com/en-us/iaas/Content/Streaming/home.htm for more information on Oracle Integration Cloud Streaming.

Chapter 2

Solution Architecture

This chapter provides an overview of the application architecture used by the integration, including:

- Solution Diagram
- Business Flows

Solution Diagram

The following diagram provides a high-level representation of the integration.
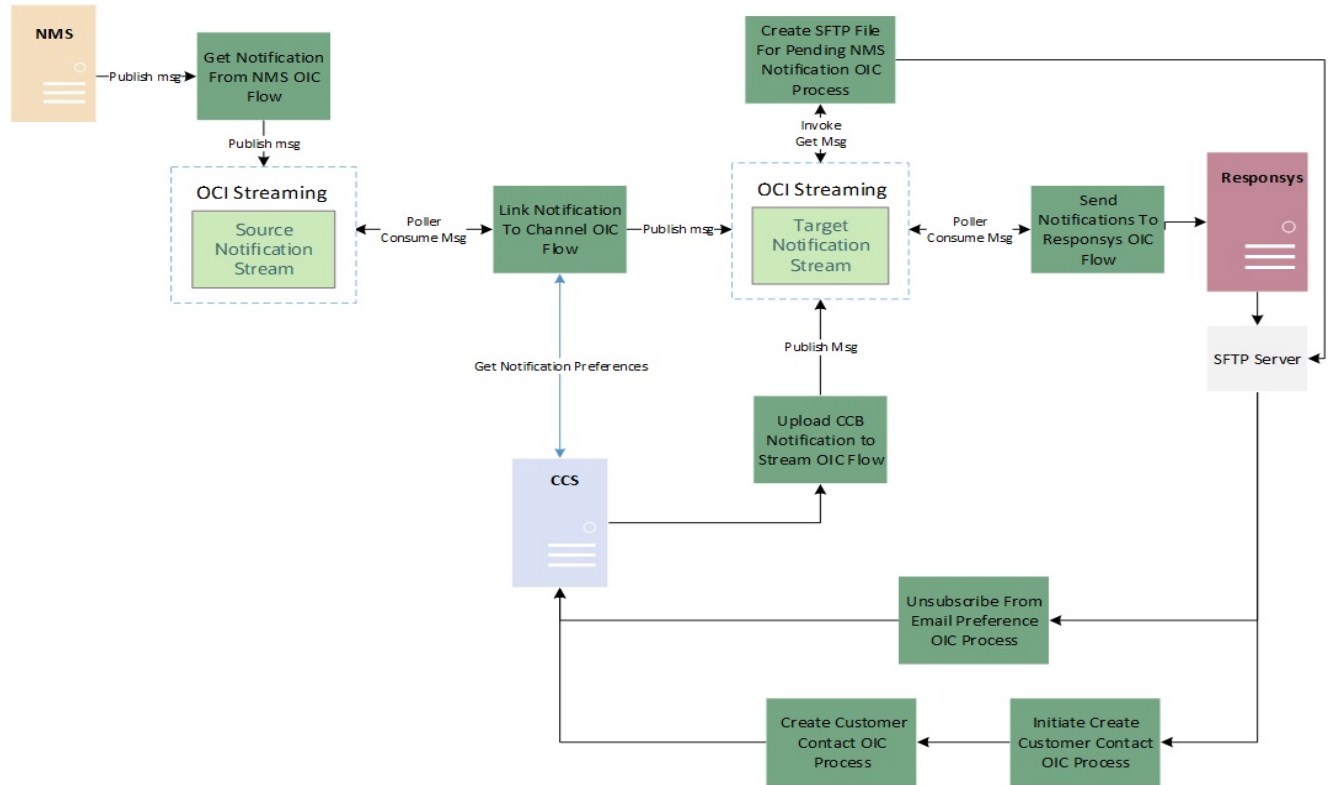
Business Flows

The integration scope supports the following business processes:

- OU OCI Stream Notif linkChannel (Oracle Integration Cloud initiated)

- OU NMS OCI Stream Notification Upload (Oracle Utilities Network Management System initiated)

- OU OCI Stream Responsys Notif Send (Oracle Integration Cloud initiated)

- OU CCB OCI Streams Notifications Upload Process (Oracle Integration Cloud initiated)

- OU Responsys CCB Pref Unsubscribe (Oracle Integration Cloud initiated)

- OU Responsys CED Process (Oracle Integration Cloud Initiated)

- OU Responsys CCB Cust Contact Create (Oracle Integration Cloud Initiated)

- Create Pending SFTP File Process (Oracle Integration Cloud Initiated)

- OU CCB Notif MetaData Sync (Oracle Integration Cloud Initiated)

Refer to the *Oracle Utilities Customer to Meter Integration to Oracle Utilities Network Management System Implementation Guide* for information on Notification Preferences Sync BPEL flow.

The documentation is available at: https://docs.oracle.com/en/industries/energy-water/integrations-index.html

OU OCI Stream Notif linkChannel (Oracle Integration Cloud initiated)

This integration process enhances the notification messages to be sent to the Oracle Responsys application.

It links the notifications coming from the source stream to the account's notification channel information coming from Oracle Utilities Customer Care and Billing. The combined information is published to the target stream to be processed by the Send Notifications to Oracle Responsys OIC process which sends the information to Oracle Responsys.

The following diagram shows a graphical representation of the Link Notification to Channel Information integration process.
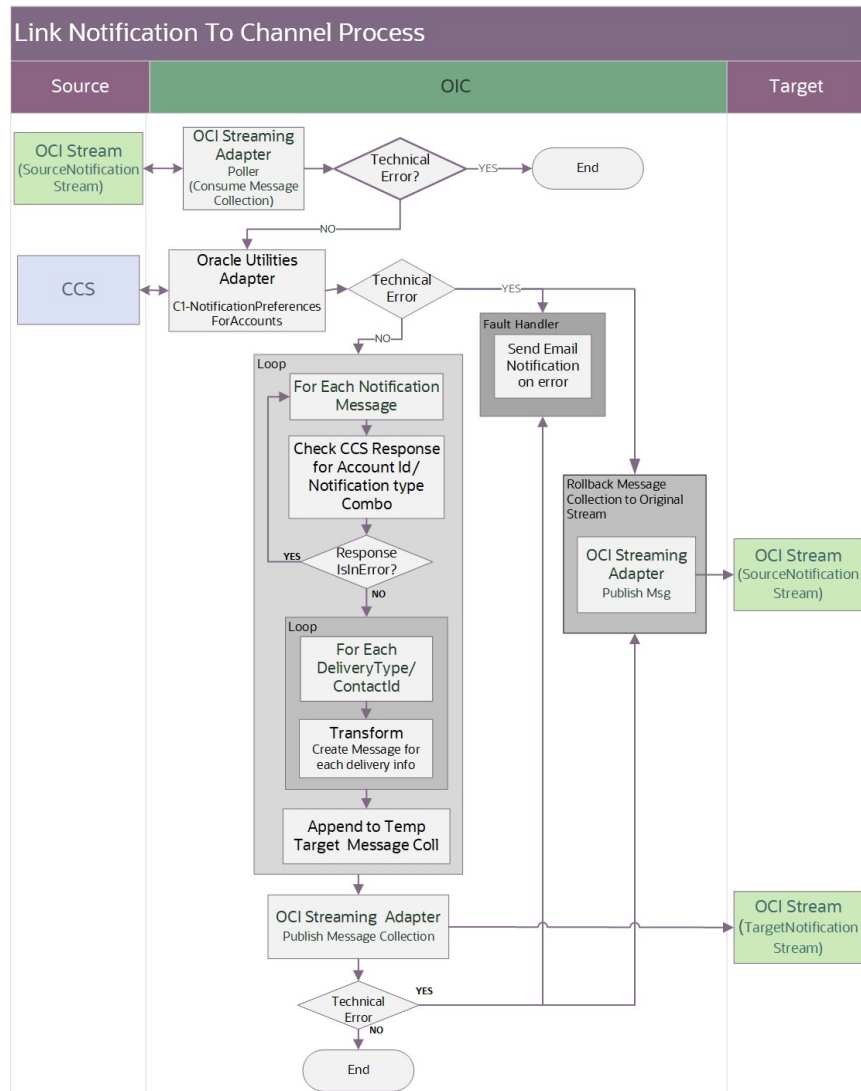
Processing Details

This integration process deployed on Oracle Integration Cloud performs the following activities:

1. Polls the Oracle Integration Cloud Stream 1 called "SourceNotification Stream". This process consumes the notification messages by batches - the maximum records fetch is 200 and it polls the source stream every sixty (60) seconds.

Note: The **Maximum Number of Records to be fetched** and **Polling Frequency** are configured on the Oracle Integration Cloud Streaming Adapter as a Trigger (Poller) at design time.

2. Invoke the CCB REST service - C1-NotificationPreferencesForAccounts to get the notification preferences for the list of account id and notification type combination provided. This input list is the list of account Id and notification type combination coming from the messages consumed in the source stream.

3. For each message fetched from Oracle Integration Cloud Source Stream, do the following:

   • Find the corresponding **Account Id** and **Notification Type** record in the Oracle Utilities Customer Care and Billing response.

   • If the response for that account ID and notification type combination record:

     • Is in error, skip the notification message and process the next notification message.

     • Most of the time Oracle Utilities Customer Care and Billing returns an error response if the account ID and notification type combination is not found in Oracle Utilities Customer Care and Billing. This scenario will be encountered when the account's notification preference has changed, i.e. customer remove subscription to a notification type, and that information has not yet been synced to Oracle Utilities Network Management System.

     • No error, proceed to the next step.

   • For each delivery type and contact ID record in the collection:

     • Transform the notification message and Oracle Utilities Customer Care and Billing response record to the format needed by the Oracle Integration Cloud Stream 2 (TargetNotification Stream).

       Note: Refer to Chapter 10: Sample Messages to understand what message is published to the target stream.

   • Append to a temporary target message output collection.

4. When all the notification messages coming from the Source stream are processed, invoke Oracle Integration Cloud Streaming adapter to publish the notification messages in batches to Oracle Integration Cloud Target Stream.

5. Error Handling for this integration process. Email notification is sent for any errors encountered in Oracle Integration Cloud. Email notifications are optional but by default it is enabled.

   Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

   Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process:

Artifacts	Value
Integration Process Name	OU OCI Stream Notif linkChannel
Integration Package Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000
OCI Streaming Adapter	• Poller to Consume Message • Operation: Consume messages from Stream • Stream: Source Notification Stream • ConsumerGroup: SourceNotificationGroup • Partition: Default • Polling Frequence: 60s • Maximum Number of records to Fetch: 200 • Message Structure: JSON • Publish to Stream • Operation: Publish messages to Stream • Stream: Target Notification Stream • Partition: Default • Message Structure: JSON Connection Name: OU OCI Streaming for Notif Hub
CCB REST IWS (REST Adapter)	Web Service Name: C1-NotificationPreferencesForAccounts Computed URL: https://{host}:{port}/{tenant}/{domain}/ccs/rest/apis/customer/v-model/notificationPreferencesForAccounts Method: POST URI: / Connection Name: • OU REST CCB for Notif Hub • CCBDirectRESTNotifPref
OIC Lookup	OUTL-BRT-NOTIFHUB_ConfigProps OUTL-BRT-NOTIFHUB_Email_ID OUTL-BRT-NOTIFHUB_DeliveryRouting OUTL-BRT-NOTIFHUB_Timezones

OU NMS OCI Stream Notification Upload (Oracle Utilities Network Management System initiated)

This integration process is triggered by Oracle Utilities Network Management System notification messages. It transforms the JSON message and publishes to the Oracle Cloud Integration Source stream. The Oracle Utilities Network Management System notification message can have more than one notification having one or more customers

(aaccounts). Each notification is transformed in one message for each customer having two data categories - header and optional data. The accountId, notificationType, messageId, origin, and notificationTemplate are part of header and data with key value pair is part of the data section.

> **Note**: Refer to Chapter 10: Sample Messages for a sample incoming JSON message received from Oracle Utilities Network Management System and sample JSON message output to understand what message is published to the target stream.

The following diagram shows a graphical representation of the Get Outage Notification from Oracle Utilities Network Management System integration process.
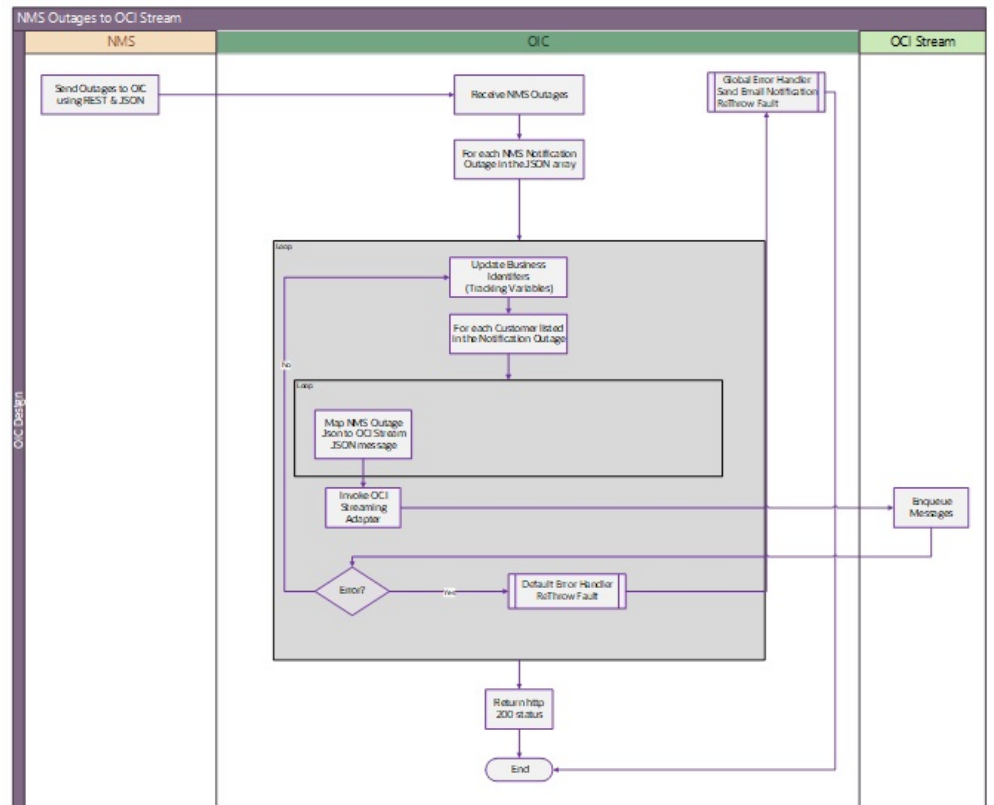
Processing Details

This integration process deployed on Oracle Integration Cloud performs the following activities:

1. The integration is triggered by Oracle Utilities Network Management System. A configuration in Oracle Utilities Network Management System has to be made to send the notifications to Oracle Integration Cloud. For more details, refer to Chapter 4: Configuring Oracle Utilities Network Management System for configuring Oracle Utilities Network Management System to send notifications.

2. Oracle Utilities Network Management System stores notification in JSON format in MYC_CSS_PARAM_VIO_TEXT in the database table. The column VIO_TEXT contains the actual JSON message.

3. The column CSS_NOTIFIED is set to 'N' by Oracle Utilities Network Management System to pick up the notification records to be processed.

4. Once the integration receives message from Oracle Utilities Network Management System notification, the JSON notification data is mapped/transformed to the OCI stream message. A notification message for each account grouping different addresses is generated and enqueued in OCI stream.

5. OCI stream requires the JSON in header and optional data format.

6. The JSON header information holds the following information:

   • accountId

   • notificationType

   • messageId

   • origin

   • notificationTemplate

   The origin is defaulted to "NMS" as the notification is generated by Oracle Utilities Network Management System system. The notificationTemplate is defaulted to empty and the field is populated after the Get Notification preference process is completed in the OU OCI Stream Notif linkChannel flow.

   Sample:

```
{
"accountId": "6386568150",
"notificationType": "ZZ-UOUT",
"messageId": "2088",
"origin": "NMS",
"notificationTemplate": ""
}
```

7. The JSON "data" information is collection of key and value pairs. By default, Cause, CrewETA, Outagetime, EstRstTime, CompletionTime and CustomerOut are populated and provided as base. In addition, data elements under PICKLIST, AFFECTEDCUSTOMERLIST(CUSTOMERADDRESS1,2..) are also populated as key value pairs as part of base only if PICKLIST is populated by the user in Oracle Utilities Network Management System, else it will be empty. The lookup OUTL-BRT-NOTIFHUB_ConfigProps holds the default property list as below:

   Lookup Name: OUTL-BRT-NOTIFHUB_ConfigProps

PropertyName	Value
nms.datafield.propertylist	OUTAGETIME,CUSTOMERSOUT,CAUSE,ESTRESTT IME,CREWETA,COMPLETIONTIME

8. After the transformation, the JSON payload is generated as per the following format:

```
{
"MessageInCollection": {
    "Message": [
    {
    "request-wrapper": {
    "content": {
    "accountId": "2200639",
    "notificationType": "INT_ S_UPLN_OUT_RESTORE_NOTIF",
    "messageId": "9005",
    "origin": "NMS",
    "notificationTemplate": "",
```

```
"data": [
{
"key": "CUSTOMERADDRESS1",
"value": "Address1"
},
{
"key": "OUTAGETIME",
"value": "1707409180"
},
{
"key": "COMPLETIONTIME",
"value": "1707409334"
},
{
"key": "ESTRESTTIME",
"value": "1707412861"
},
{
"key": "CUSTOMERSOUT",
"value": "5"
},
{
"key": "CAUSE",
"value": "TROUBLE_CALL"
},
{
"key": "CREWETA",
"value": "0"
},
{
"key": "ACTION_TEXT",
"value": ""
},
{
"key": "ADV_ENVIRON_OM",
"value": "Unselected"
},
{
"key": "ADV_WEATHER_OM",
"value": "Unselected"
},
{
"key": "BREAKER_TEXT",
"value": ""
},
{
"key": "CAUSE_OM",
"value": "Primary Mesh"
},
{
"key": "CONTROL_ZONE_TEXT",
"value": "OPAL  Stark  Lake  Lake Sub  2414"
},
{
"key": "DEF_EQUIP_OM",
"value": "Customers Equipment"
},
{
"key": "DEVICE_TEXT",
"value": "T11449"
},
{
```

```
"key": "DISTRICT_TEXT",
"value": ""
},
{
"key": "DIVISION_TEXT",
"value": ""
},
{
"key": "EMERG_SW_ORDER_TEXT",
"value": ""
},
{
"key": "EST_OUT_OM",
"value": ""
},
{
"key": "FAILURE_OM",
"value": "Unknown"
},
{
"key": "FEEDER_TEXT",
"value": ""
},
{
"key": "FOREIGN_INTERF_OM",
"value": "Unselected"
},
{
"key": "HUMAN_ELEM_OM",
"value": "Unselected"
},
{
"key": "INTERRUPT_DEV_OM",
"value": "Substation Breaker"
},
{
"key": "OFFICE_TEXT",
"value": ""
},
{
"key": "OTHER_CAUSE_OM",
"value": "Overload"
},
{
"key": "PRIMARY_CAUSE_OM",
"value": "Foreign Interference"
},
{
"key": "REMEDY_OM",
"value": "Equipment Removed"
},
{
"key": "SCHEDULED_DEV_OM",
"value": "Unselected"
},
{
"key": "SYSTEM_OM",
"value": "Transmission OH"
},
{
"key": "TRANS_TEXT",
"value": ""
```

```
},
{
"key": "TYPE_OM",
"value": "Unselected"
},
{
"key": "VEGETATION_OM",
"value": "Unselected"
}
]
}
},
"config-parameters": {
"key": "INT_ S_UPLN_OUT_RESTORE_NOTIF"
}
```

9. The integration then publishes the notification JSON to OCI source stream. The transaction is marked by Oracle Utilities Network Management System as success by updating the CSS_NOTIFIED='Y' at the end of the integration execution.

10. On technical or business errors, such as invalid JSON message, OCI Stream unavailable, an email is sent with the error details, including Instance ID and error fault details.

    Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

    Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* included in this release about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process.

Artifacts	Value
Integration Process Name	OU NMS OCI Stream Notif Upload
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.1.24.1000
Identifier	OUTL-NMS_STREAM_NOTIF_UPL
Oracle Utilities Adapter	Get publish JSON message definition from NMS. Operation: publish Message format: JSON Connection name: OU REST NMS for Notif Hub
OCI Streaming Adapter	Publish message to source OCI stream. Operation: Publish messages to Stream Stream: Source Notification Stream Partition: Default Message Structure: JSON Connection Name: OU OCI Streaming for Notif Hub

Artifacts	Value
OIC Lookup	OUTL-BRT-NOTIFHUB_ConfigProps
	OUTL-BRT-NOTIFHUB_Email_ID
	OUTL-BRT-NMS_STREAM_TROUBLE_CODE
	OUTL-BRT-NOTIFHUB_NotifMetaData

OU OCI Stream Responsys Notif Send (Oracle Integration Cloud initiated)

This integration process reads the final transformed notification messages from different application in the OCI Stream and sends to Oracle Responsys to be delivered via email or SMS.

The following diagram shows a graphical representation of the Send Notifications to Oracle Responsys process.

Processing Details

Integration Properties	Description	Initial Value
responsysInstance	This is the responsysInstance representing an Oracle Responsys instance. It is used to sort the messages read from the Target Notification Stream, belonging to the Oracle Responsys instance linked to this particular OIC instance. The lookup table OUTL-BRT-NOTIFHUB_DeliveryInstance is used to translate responsysInstance to a value that is populated in the email notifications to identify the Oracle Responsys instance.	

The integration process includes the following activities:

1. This Oracle Integration Cloud process reads the final transformed notification messages dropped in the TargetNotificationStream through OCI Streaming Adapter poller. The messages are read in batches and assigned to "ConsumedNotifCollection" global variable through data stitch.

2. For each message in the ConsumedNotifCollection:

   a. Assign the deliveryChannel, notificationTemplate from the first notification message and responsysInstance(Integration Property) to variables for sorting.

   b. Iterate over messages in the ConsumedNotifCollection.

   c. Sort the messages. If the messages match the sort parameters for deliveryChannel, notificationTemplate and responsysInstance(Integration Property), copy them to the SortedNotifCollection global variable.

   d. If the messages do not match the sort parameters, copy them to the UnprocessedNotifCollection global variable.

   e. The messages in SortedNotifCollection are checked for the Oracle Responsys instance by examining the deliveryInstance field.

   f. The grouped messages in the SortedNotifCollection are transformed to Oracle Responsys Email or SMS format based on the deliveryChannel of the group.

   g. Oracle Responsys API for the particular campaign identified by notification template is invoked.

   h. The error thrown during API invocation is handled by respective fault handlers.

   Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

   Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* about mapping information for data elements.

Technical Details

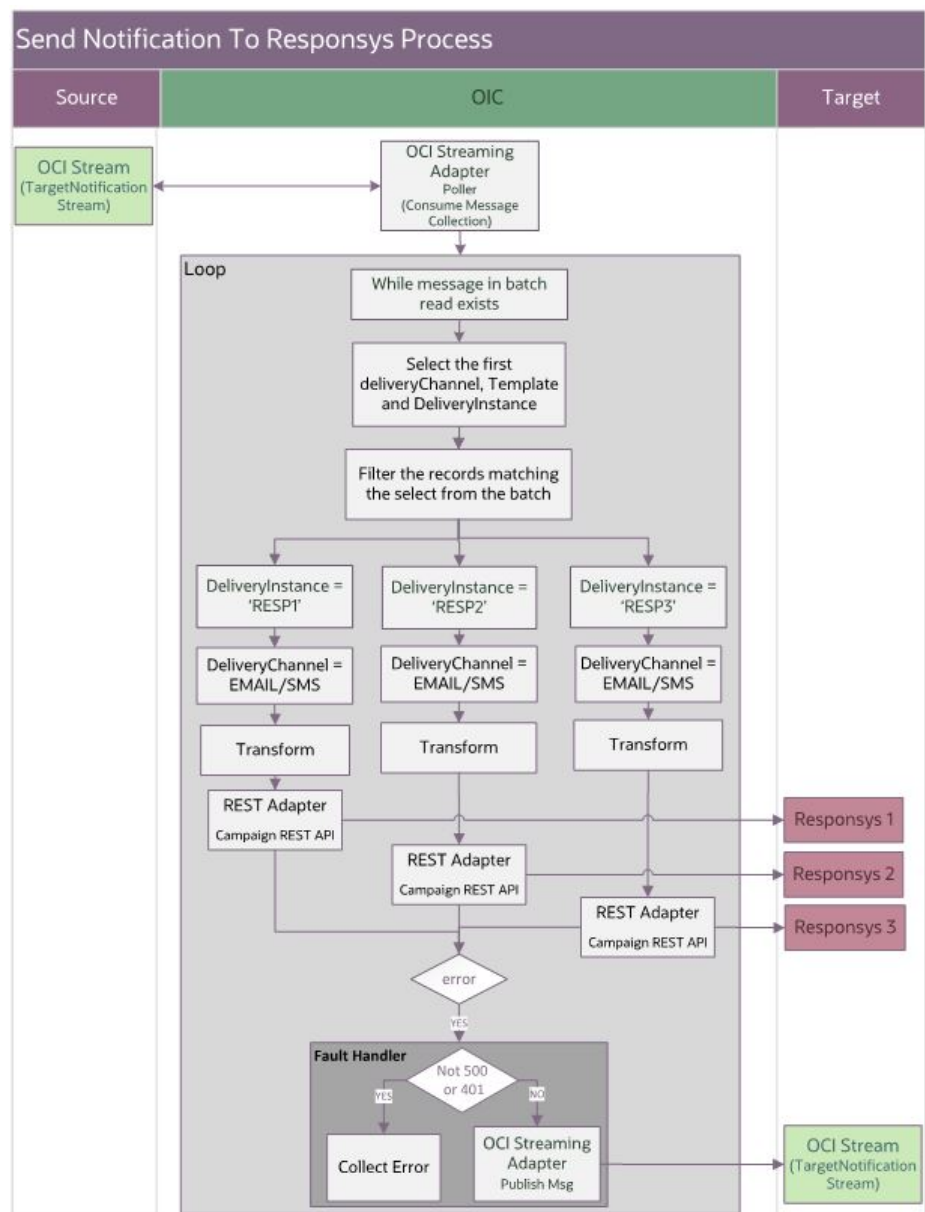The following table describes the integration processes and the respective artifacts used in this integration process.

Artifacts	Value
Integration Process Name	OU OCI Stream Responsys Notif Send
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000

OCI Streaming Adapter	• Poller to Consume Message
	
	  • Operation: Consume messages from Stream
	  • Stream: TargetNotificationStream
	  • Consumer Group: NotificationGroup
	  • Partition: Default
	  • Polling Frequence: 60 s
	  • Maximum Number of records to Fetch: 200
	  • Message Structure: JSON
	
	• Publish to Stream
	
	  • Operation: Publish messages to Stream
	  • Stream: TargetNotificationStream
	  • Partition: Default
	  • Message Structure: JSON
	  • Connection Name: OU OCI Streaming for Notif Hub

Responsys Web Service	WebService: /rest/api/v1.3/campaigns/{notificationTemplate}/email /rest/api/v1.3/campaigns/{notificationTemplate}/sms Method: POST Connection Name: OU REST Responsys for Notif Hub
OIC Lookup	OUTL-BRT-NOTIFHUB_ConfigProps OUTL-BRT-NOTIFHUB_Email_ID OUTL-BRT-NOTIFHUB_DeliveryChannel OUTL-BRT-NOTIFHUB_DeliverynInstance OUTL-BRT-NOTIFHUB_Timezones

OU CCB OCI Streams Notifications Upload Process (Oracle Integration Cloud initiated)

This integration process is used to process Oracle Utilities Customer Care and Billing notifications as outbound messages from Oracle Utilities Customer Care and Billing and publish it into Oracle OCI Streams. The information is sent synchronously from Oracle Utilities Customer Care and Billing, regardless of the notification type, and OIC sends back a synchronous response. Oracle Utilities Customer Care and Billing does not use the externalId sent as part of the response.

The following diagram shows a graphical representation of the CCB Notifications to OCI Stream process.
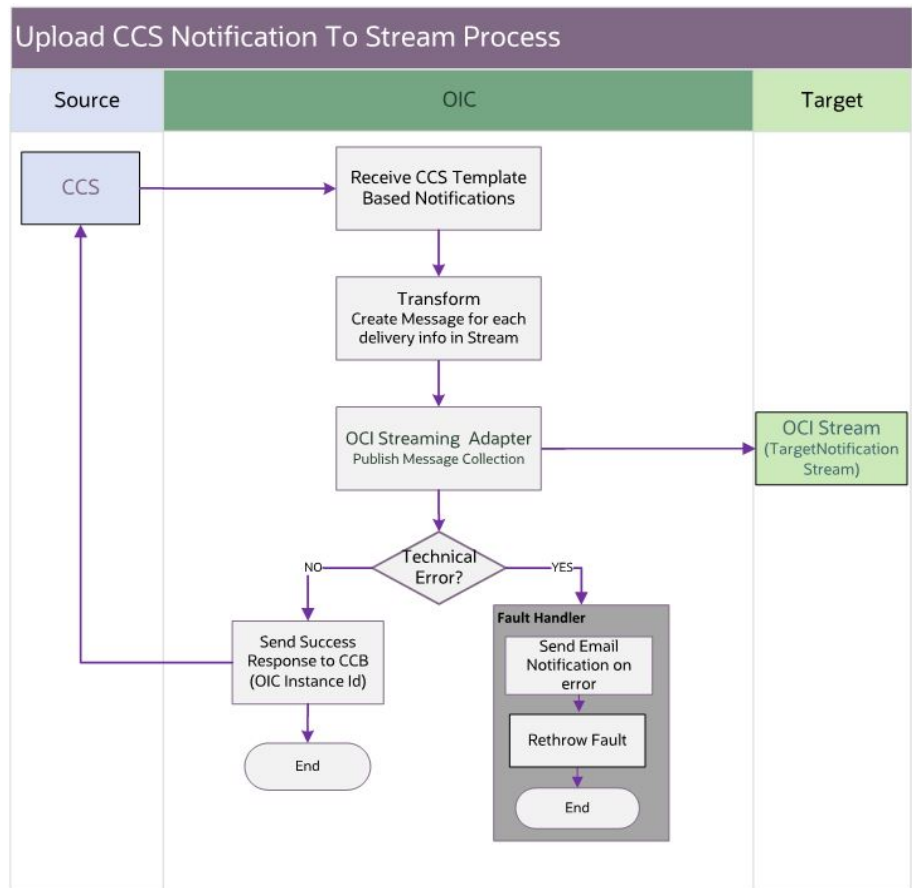
Processing Details

The integration process includes the following activities:

1. Oracle Utilities Customer Care and Billing sends the template based outbound notification message to the Oracle Utilities CCB OCI Stream Notif Upload integration process deployed on Oracle Integration Cloud.

2. The Oracle Utilities CCB OCI Stream Notif Upload integration process transforms the outbound message from Oracle Utilities Customer Care and Billing to the standard message format of the OCI Target Notification Stream. There is a separate notification message created for each delivery info going into OCI Stream. The messages are published in bulk into OCI Target Notification Stream using the OCI Stream Adapter based on the Notification Type priority.

3. In case of error, the integration will send a notification email to the recipient "to" configured in the OUTL-BRT-NOTIFHUB_Email_ID dvm.

4. In case of successful batch write to the OCI Streams the OIC instance ID is returned to Oracle Utilities Customer Care and Billing.

    Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

    Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* included in this release about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process.

Artifacts	Value
Integration Process Name	OU CCB OCI Stream Notif Upload
Identifier	OUTL-BA-CCB_STREAM_NOTIF_UPLOAD
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000
CCB BO	C1-TMPLNTFRQ
Oracle Utilities Adapter Connection	OU REST CCB for Notif Hub

Oracle Streams Adapter	OUTL-BRT-STREAM-NOTIF_HUB
	
	• Publish to Stream
	  • Operation: Publish messages to Stream
	  • Stream: Target Notification Stream
	  • Partition: Default
	  • Message Structure: JSON

OIC Lookup	OUTL-BRT-NOTIFHUB_ConfigProps OUTL-BRT-NOTIFHUB_Email_ID OUTL-BRT-NOTIFHUB_DeliveryInstance OUTL-BRT-NOTIFHUB_NotifMetaData

OU Responsys CCB Pref Unsubscribe (Oracle Integration Cloud initiated)

This scheduled integration process unsubscribes a customer's email from a specific notification type. The integration calls an Oracle Utilities Customer Cloud Service API which in turn inactivates the email notification preference (Unsubscribe) for transactional messages.

The following figure shows the representation of a user flow:
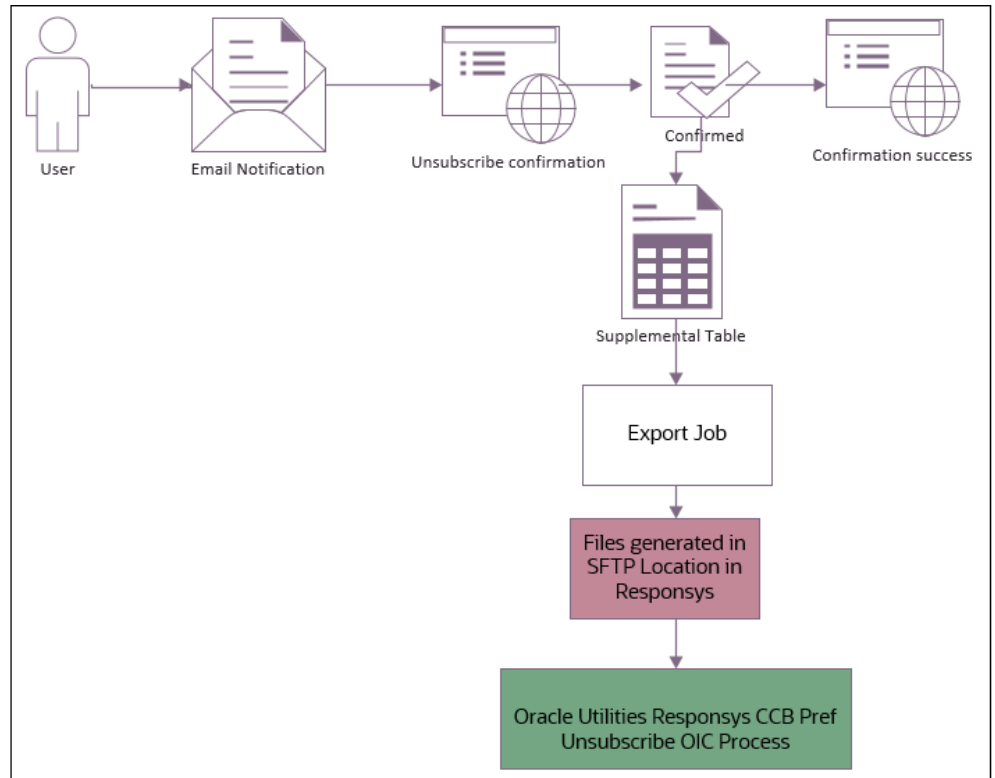
1. User clicks the "Unsubscribe" link in the notification email received.

2. User confirms the Unsubscribe action on the confirmation page.

3. The information required for unsubscription is stored in the Oracle Responsys supplemental table.

4. An export job to export supplemental table data is scheduled in Oracle Responsys.

5. This job exports the data stored in the supplemental table to a csv file and stores the file in the Oracle Responsys SFTP location.

6. Oracle Utilities Responsys CCB Pref Unsubscribe process is scheduled to read and process these files in the SFTP location.

The following diagram shows a graphical representation of the process.
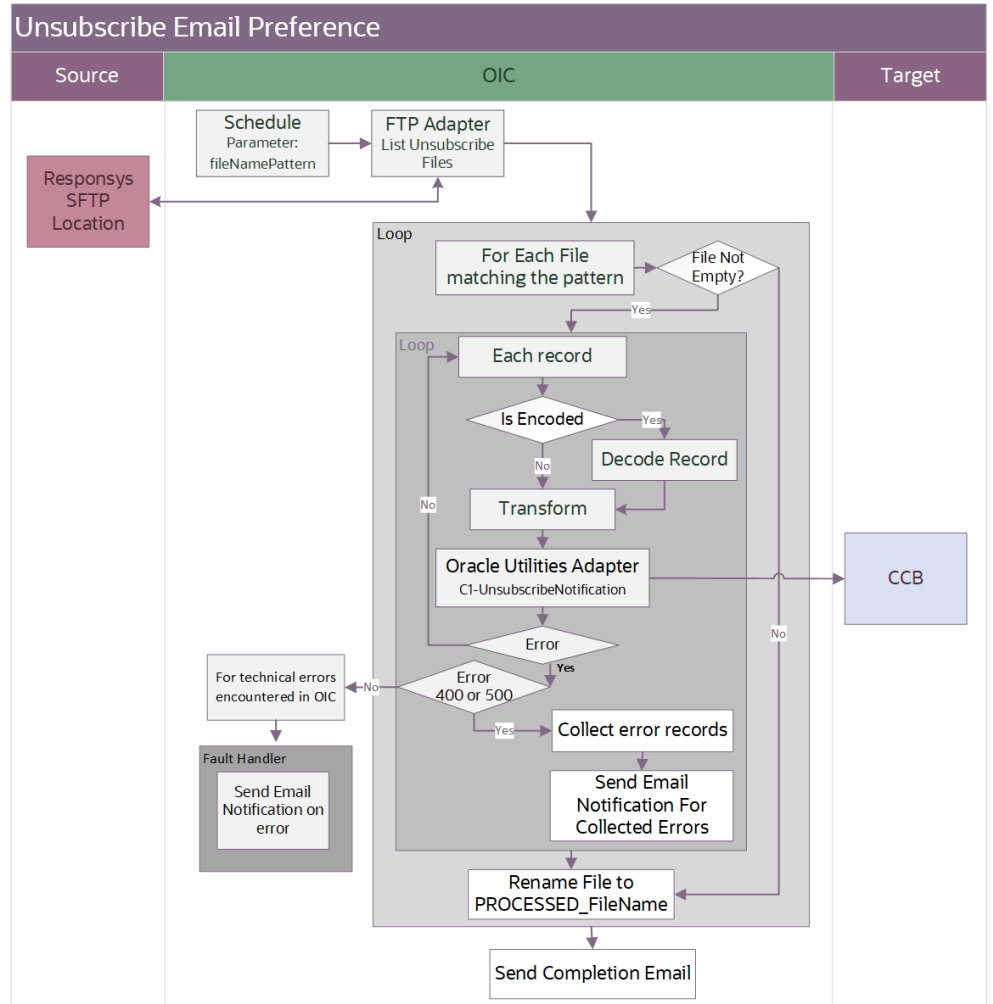
Processing Details

This is a scheduled integration process deployed on Oracle Integration Cloud that performs the following activities:

1. This process is triggered by a schedule and has one schedule parameter. Define the integration schedule for this process according to your business requirement.

Integration Properties	Description	Initial Value
fileNamePattern	This is the filename pattern to use to read the unsubscribe files from the Oracle Responsys SFTP location. File name pattern may satisfy a single file or multiple files.	Unsubscribe*

2. Invoke the FTP Adapter to list the unsubscribed files in the Oracle Responsys SFPT server given the filename Pattern criteria.

3. If the filelist is not empty:

   • Yes: Go to step 4.

- No: Rename the file name to prefix the "PROCESSED-" in front of the filename. And move on to the next file, if any.

4. For each unsubscribe record in the file, the process does the following:

   - Check if the encoded flag is true, decode the record.

   - Transform the unsubscribe information received from Oracle Responsys to the CCS Rest service format.

   - Invoke the CCS REST service - C1-UnsubscribeNotification to unsubscribe a customer's email from a notification type.

   - If the CCS REST API returns:

     - Status 200 (OK), process the next record in the file.

     - Error status 400 or 500, add the record in error to a temporary error collection. Then, it continues to process the next job record.

     - Error status NOT 400 or 500, stop the process and send out an error email notification from the global fault handler. These errors are considered technical or remote faults (Sample Error Status: 401, 404, 503, etc.)

   - Check if the error count in the temporary errorcollection reached the value for property name maxerrorcount.per.emailnotification defined in the configuration properties lookup.

     - If yes, send an error email notification with the unsubscribe records in error.

     Note: This property max.error.count determines the number of errors to be sent out in an error email notification and how frequent an error email notification is sent out.

Integration Properties	Description	Initial Value
ResponsysInstanceId	This is the ResponsysInstanceId representing a Responsys instance. The lookup table OUTL-BRT-NOTIFHUB_DeliveryInstance is used to translate the ResponsysInstanceId to a value that is populated in the email notifications to identify the Responsys Instance which is generating errors.	RESP1

5. If the processed file has no errors, rename the .zip and .txt files to prefix them with "PROCESSED-".

6. After all records in the files are processed, send a process completed email notification with the name of the file processed, record processed count, records in error count, if any and the total records.

7. Error Handling for this integration process. Email notification is sent for any errors encountered in Oracle Integration Cloud. Email notifications are optional but by default it is enabled.

   Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

   Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* included in this release about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process.

Artifacts	Value
Integration Process Name	OU Responsys CCB Pref Unsubscribe
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000
CCB BO	C1-TMPLNTFRQ
Oracle FTP Adapter	• Operation: List Files • Input Directory: / • File Name Pattern: * • Max Files: 100 • Minimum Age: 0 seconds • List Files Recursively • Operation: Read File • Operation: Download File • Operation: Move File (Rename to PROCESSED"_fileName) • Connection Name: OU SFTP for Notifican Hub
CCB REST IWS (Utilities Adapter)	Web Service Name: C1- UnsubscribeNotifications Computed URL: https://{host}:{port}/{tenant}/{domain}/ccs/rest/apis/customer/v-model/C1UnsubscribeNotifications
OIC Lookup	Method: POST URI: / Connection Name: • OU REST CCB for Notif Hub • OUTL-BRT-NOTIFHUB_ConfigProps • OUTL-BRT-NOTIFHUB_Email_ID • OUTL-BRT-NOTIFHUB_DeliveryChannel • OUTL-BRT-NOTIFHUB_DeliveryInstance

OU Responsys CED Process (Oracle Integration Cloud Initiated)

This integration process picks the CED file, processes it and sends a request to create Customer Contact in Oracle Utilities Customer Care and Billing.

The following diagram shows a graphical representation of the Create Customer Contact process:
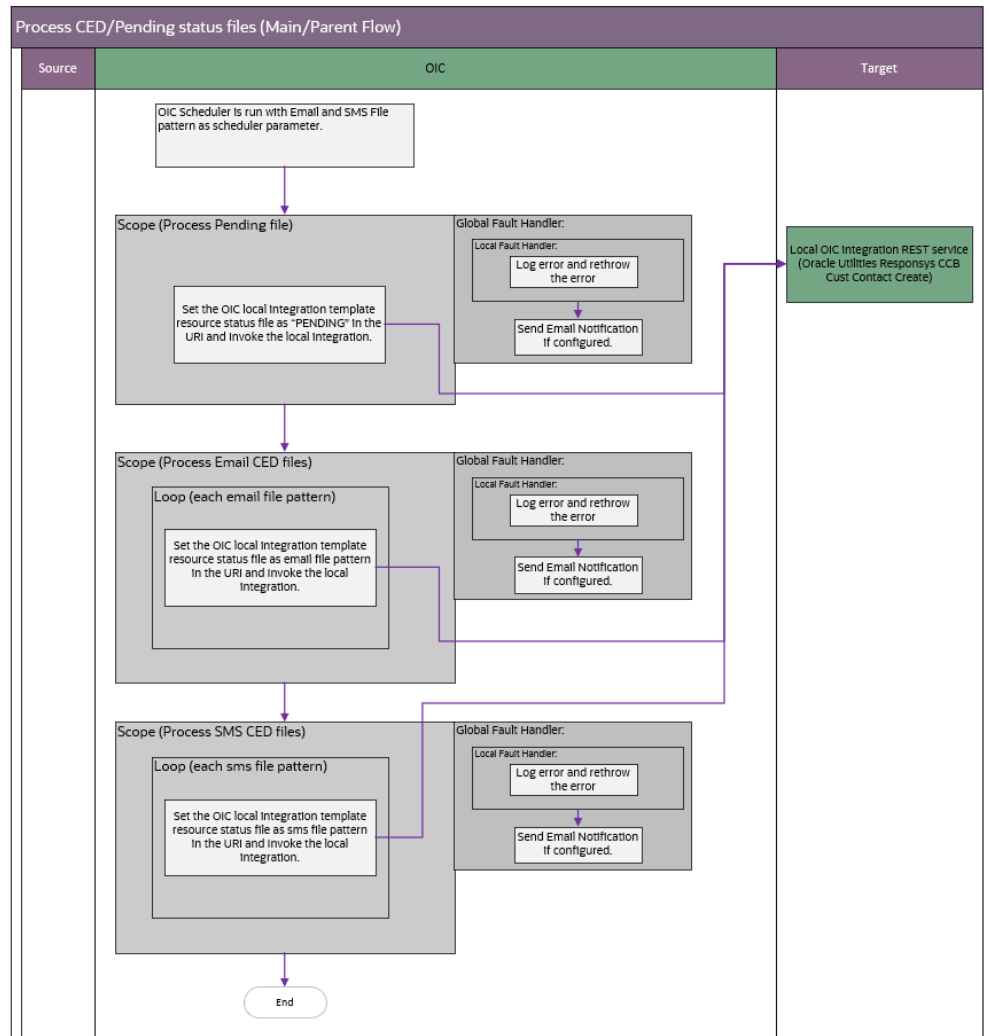
Processing Details

This is a scheduled integration process deployed on Oracle Integration Cloud that performs the following activities:

1. This process is triggered by a schedule and has one schedule parameter. Define the integration schedule for this process according to your business requirement.

Schedule Parameters	Description	Initial Value
EmailFilePattern	Email CED file pattern to be processed	"SENT,FAIL,SKIPPED"
SMSFilePattern	SMS CED file pattern to be processed	"SMS_SENT, SMS_FAIL, SMS_SKIPPED"

2. Responsys CED file processing is handled using two integration process. This integration is the main process, and the child process is "Oracle Utilities Responsys

CCB Cust Contact Create". The main integration flow sends file pattern to the child integration flow using the REST API call.

3. Capture the file pattern and if the file pattern is "PENDING" as per scheduled parameters, call the child integration (Oracle Utilities Responsys CCB Cust Contact Create) and mention the resource path as CEDPending/PENDING. The child integration in invoked and processes the "PENDING" status files.

4. Capture the file pattern and if the file pattern is "EmailFilePatten", call the child integration (Oracle Utilities Responsys CCB Cust Contact Create) and mention the resource path as CED/emailFilePattern. The child integration in invoked and processes the email CED status files.

5. Capture the file pattern and if the file pattern is "SMSFilePatten", call the child integration (Oracle Utilities Responsys CCB Cust Contact Create) and mention the resource path as CED/smsFilePattern. The child integration in invoked and processes the sms CED status files.

6. Email notification is sent for any errors encountered in Oracle Integration Cloud. Email notifications are optional but by default it is enabled.

7. Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

8. Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* included in this release about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process.

Artifacts	Value
Integration Process Name	OU Responsys CED Process
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000
Identifier	OUTL-BA-RESP_CED_PROCESS
Local OIC Integration (Invoking PENDING status file)	• OIC integration name: OU Responsys CCB Cust Contact Create • API path: oicinstance/ic/api/integration/v1/flows/rest/ • OUTL-BA-RESP_CCB_CC_CREATE /1.0/metadata • Resource path: /CEDPending/{statusFile} • Response media type: application/json • Operation: processPendingFile • Http Method: POST • Connection Name: OU SFTP for Notif Hub

Artifacts	Value
Local OIC Integration (Invoking EMAIL status file)	• OIC integration name: OU Responsys CCB Cust Contact Create • API path: oicinstance/ic/api/integration/v1/flows/rest/ • OUTL-BA-RESP_CCB_CC_CREATE /1.0/metadata • Resource path: /CED/{statusFile} • Response media type: application/json • Operation: processCEDFile • Http Method: POST. • Connection Name: OU SFTP for Notif Hub
Local OIC Integration (Invoking SMS status file)	• OIC integration name: OU Responsys CCB Cust Contact Create • API path: oicinstance/ic/api/integration/v1/flows/rest/ • OUTL-BA-RESP_CCB_CC_CREATE /1.0/metadata • Resource path: /CED/{statusFile} • Response media type: application/json • Operation: processCEDFile • Http Method: POST • Connection Name: OU SFTP for Notif Hub
OIC Lookup	• OUTL-BRT-NOTIFHUB_ConfigProps • OUTL-BRT-NOTIFHUB_Email_ID

OU Responsys CCB Cust Contact Create (Oracle Integration Cloud Initiated)

This integration creates customer contact in Oracle Utilities Customer Care and Billing by picking up the Oracle Responsys CED and pending files from Oracle Responsys SFTP location. The integration is the sub/child process and will be initiated by its main/parent integration service called "Oracle Utilities Responsys CED Process". The Oracle Utilities Responsys CED Process integration will invoke this integration multiple times with different file pattern in a single run. The main/parent process sends request to this integration process with CED EMAIL/SMS and pending file pattern.

The following diagram shows a graphical representation of the Create Customer Contact process.
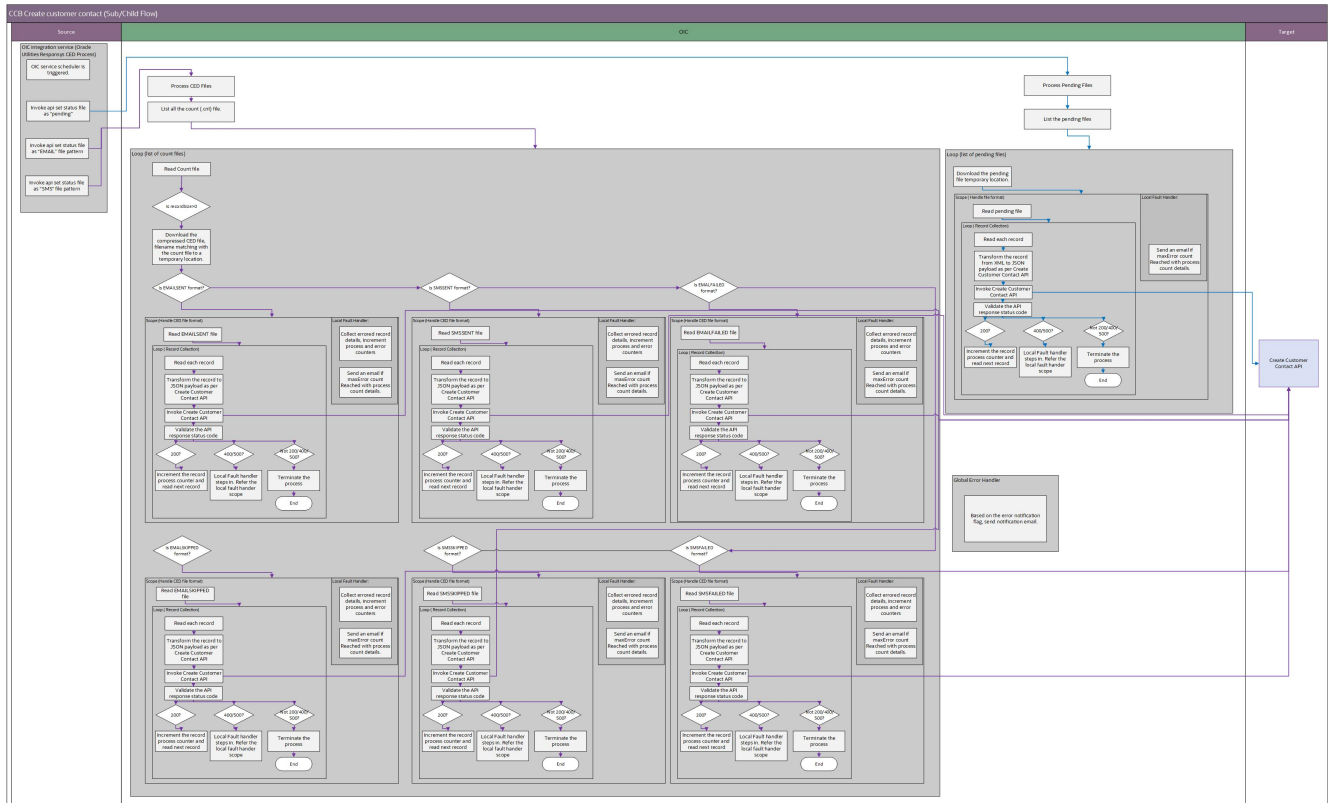
Processing Details

This is a scheduled integration process deployed on Oracle Integration Cloud that performs the following activities:

1. This process is triggered by a schedule parameters.

Integration Parameters	Description	Initial Value
pendingTargetDirectory	SFTP Directory location where the pending files should be created and placed.	download/pending
pendingSourceDirectory	SFTP Directory location where the pending files needs to be picked for file processing.	download/pending
ResponsysAcctId	Responsys account ID	81290
sourceDirectory	Responsys CED file's SFTP folder location. From this location, the integration process will pick the files and process further.	download/ced

Integration Parameters	Description	Initial Value
targetDirectory	CED processed file's SFTP folder location. The integration will place the processed files in the target directory and will mark as processed by prefixing the file name with "PROCESSED".	download/ced
ResponsysInstanceId	Responsys instance ID	RESP1
ResponsysListId	Responsys List ID	

2. This integration is the sub/child process and will be called by its main/parent integration service called "OU Responsys CED Process".

3. Once the integration is initiated by "OU Responsys CED Process", capture the file pattern to process. The integration can process the pending, sms, and email CED file patterns.

4. The integration has two main flows, one is process CED files which included EMAIL AND SMS both and the second one is to process "PENDING" status files.

5. The processCEDFile (EMAIL and SMS file pattern) flow is as below:

   • Oracle Responsys generates compressed CED file in zip format and generates a .cnt file as well with same file name format. The .cnt file contains information about the record size present in compressed CED file.

   • List all the count (.cnt) file.

   • Loop through each count file and check if the record count in not zero.

   • Mark the .cnt file and the compressed CED file as processed by adding "PROCESSED" before the file name if the record count is zero.

   • Pick the compressed CED file and unzip the compressed CED file in a temporary location if the record count is not zero for further processing.

   • Read the unzipped file (CED txt file), transform the content based on CED status file for example, SENT, CLICKED, FAILED, SKIPPED to json payload to call CCB API to create customer contact.

   • The record is filtered based on "ResponsysListIds" integration property provided and the isCustomerContactRequired flag from OUTL-BRT-NOTIFHUB_NotifMetaData is checked and based on the outcome, the record is sent to create customer contact. To summaries, the above business rules are applied before calling the CCB create customer contact.

   • Provide External Identifier and Notification System Identifier and call the CCB API to create customer contact.

   • If the CCB API call returns the HTTP status code as 200 (OK), process the next record in the file.

   • If HTTP status code 400 or 500 is received, capture the record in error to a temporary error collection and proceed further to process the next job record.

   • If HTTP status code is not 400 or 500, stop the process and send out an error email notification from the global fault handler. These errors are considered technical or remote faults (Sample Error Status: 401, 404, 503, etc.)

- On error scenarios specially 400 or 500 http status code:
  - Log error on every error occurred.
  - Capture the failed records. This can be later used to send out an email to have the file process status.

6. The processPendingFile ("PENDING" file pattern) flow is as below:

- The pending status xml file is generated in a separate process by Oracle Utilities Responsys SFTP File Create integration service. The count .cnt file is not generated in this process.

- List all the pending file.

- Loop through each pending file, download in a temporary location and process each xml message.

- Prepare and transform the xml message to json message.

- Mark the flag called customerContactIdentifier with Originating System Identifier and sourceOriginatingId in the JSON payload.

- Invoke the CCB Create Customer Contact API and pass the Json payload.

- On success API call, CCB Create Customer Contact API will return HTTP status code 200. On success response, rename the pending file as processed by prefix with "PROCESSED" and send a completion email with details.

- On error, send an uncompleted process email with details. As part of error handler, if HTTP status code is 400 or 500, capture the record in error to a temporary error collection and proceed further to process the next job record. Using maxErrorCount counter variable, an error email will be sent and the tempMaxErrorCount temporary variable will reset back to 0 until its reach the maxErrorCount.

- If HTTP status code is not 400 or 500, stop the process and send out an error email notification from the global fault handler. These errors are considered technical or remote faults (Sample Error Status: 401, 404, 503, etc.)

7. Email notification is sent for any errors encountered in Oracle Integration Cloud. Email notifications are optional, but by default it is enabled.

Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* included in this release about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process.

Artifacts	Value
Integration Process Name	OU Responsys CCB Cust Contact Create
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000
Identifier	OUTL-BA-RESP_CCB_CC_CREATE

Artifacts	Value
Oracle Utilities Adapter	• Business Object: C1-CustomerContactNotification • Operation: syncCustomerContact • Request Payload Type: JSON • Response Payload type: JSON • API type: REST • API url: https://{domain}:port/ouaf/rest/ouaf/openapi/iws/catalog • Http Method: POST. • Connection Name: OU REST CCB for Notif Hub
Oracle FTP Adapter	• Endpoint name: GetAllTheCountFile • Endpoint description: Get the list of count files. • Operation: List File • Input Directory: /tmp • File Name Pattern: *.cnt • Max Files: 100 • Minimum Age: 0 seconds • List Files Recursively: Checked • Ignore File Permissions: Unchecked • Connection Name: OU REST CCB for Notif Hub
Oracle FTP Adapter	• Endpoint name: ReadCountFile • Endpoint description: Read the count file • Operation: Read a file • Input Directory: Directory name from Oracle FTP Adapter (GetAllTheCountFile) • File Name: File name from Oracle FTP Adapter (GetAllTheCountFile) • Select the transfer mode: ASCII • Do you want to specify the structure for the content of file: No • Connection Name: OU SFTP for Notif Hub

Artifacts	Value
Oracle FTP Adapter	• Endpoint name: DownloadActualFile • Endpoint description: Download file with file pattern in scheduler parameters. • Operation: Download File • Input Directory: Directory name from Oracle FTP Adapter (GetAllTheCountFile) • File Name: File name from Oracle FTP Adapter (GetAllTheCountFile) • Select the transfer mode: Binary • Download Directory: /temp/resp • Perform unzip on compressed file downloaded from external FTP server: Checked • Connection Name: OU SFTP for Notif Hub
Oracle FTP Adapter	• Endpoint name: MarkCountFileAsProcess • Endpoint description: Rename the count file as processed • Operation: Move a File • Input Directory: Directory name from Oracle FTP Adapter (GetAllTheCountFile) • Input Filename: Prefixed with "PROCESSED" and current timestamp • Connection Name: OU SFTP for Notif Hub
Oracle FTP Adapter	• Endpoint name: MarkZipFileProcess • Endpoint description: Rename the CED compressed file as processed. • Operation: Move a File • Input Directory: Directory name from Oracle FTP Adapter (GetAllTheCountFile) • Input Filename: Prefixed with "PROCESSED" to the existing processed filename. • Connection Name: OU SFTP for Notif Hub
Oracle FTP Adapter	• Endpoint name: ListPendingFile • Endpoint description: List the pending xml file • Operation: List File • Input Directory: /tmp • File Name Pattern: PENDING*.xml • Max Files: 100 • Minimum Age: 0 seconds • List Files Recursively: Checked • Ignore File Permissions: Unchecked • Connection Name: OU SFTP for Notif Hub

Artifacts	Value
Oracle FTP Adapter	• Endpoint name: DownloadPendingFile
	• Endpoint description: Get all the pending files.
	• Operation: Download File
	• Select the transfer mode: ASCII
	• Input Directory: Directory name from Oracle FTP Adapter (ListPendingFile)
	• File Name: File name from Oracle FTP Adapter (ListPendingFile)
	• Download directory: /tmp/resp
	• Connection Name: OU SFTP for Notif Hub
Oracle FTP Adapter	• Endpoint name: RenamePendingFile
	• Endpoint description: Rename the processed pending files
	• Operation: Move a File
	• Directory path: /
	• File Name: *
	• Target Directory path: /
	• Target File Name: *
	• Overwrite File: Unchecked
	• Connection Name: OU SFTP for Notif Hub
OIC Lookup	• OUTL-BRT-NOTIFHUB_ConfigProps
	• OUTL-BRT-NOTIFHUB_Email_ID
	• OUTL-BRT-NOTIFHUB_CustContactStatus
	• OUTL-BRT-NOTIFHUB_OriginSystem
	• OUTL-BRT-NOTIFHUB_DeliveryInstance
	• OUTL-BRT-NOTIFHUB_NotifMetaData

Create Pending SFTP File Process (Oracle Integration Cloud Initiated)

This integration process creates pending status file from the OCI notification message. The integration is scheduled to pick messages from OCI stream, converts the JSON message to XML format and create a Pending status xml file for further processing. The integration "OU Responsys CCB Cust Contact Create" further processes each xml messages and invokes CCB Create Customer API to create a customer contact with a pending status. The job of the current integration is just to convert the OCI JSON messages to XML and write an XML file prefixed with "PENDING" to an Oracle Responsys SFTP location.

The following diagram shows a graphical representation of the Create Pending Status file.
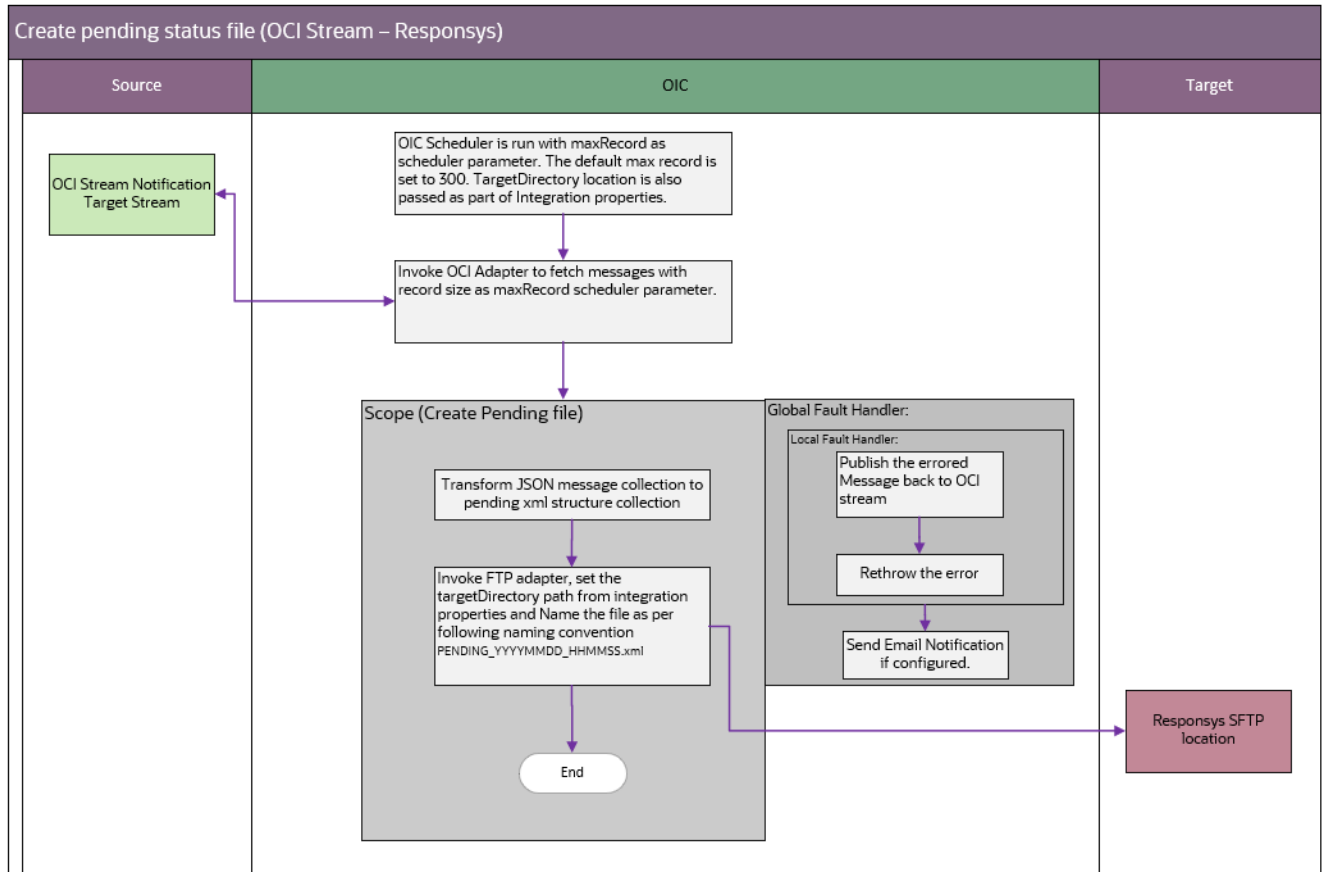
Processing Details

This is a scheduled integration process deployed on Oracle Integration Cloud that performs the following activities:

1. This process is triggered by a schedule and has one schedule parameter.

Schedule Parameters	Description	Initial Value
maxRecords	Maximum records to be fetched from the OCI Stream	300

Integration Parameters	Description	Initial Value
targetDirectory	Oracle Responsys SFTP directory location where the Pending files will be written/created.	SFTP location where Oracle Responsys is installed

2. Define the integration schedule for this process according to your business requirement.

3. Invoke OCI Stream adapter to consume the message from TargetNotificationStream and set the maxfetch records with maxRecords from schedule parameters.

4. The consumer group is set to CustomerContactGroup as part of OCI stream adapter configuration.

5. Assign the total number of messages fetch to a count variable.

6. Check the message count and if exits any messages, transform the json payload from OCI stream to XML format.

7. Invoke the FTP adapter and create a file with the following format:

   "PENDING" + current time stamp

8. A xml file with prefixed with "PENDING" and timestamp file will be created in the specified targetDirectory location, configured as part of integration parameter.

9. On any error during this process, the consumed message will be placed back to the OCI target stream for further process in next run.

10. Email notification is sent for any errors encountered in Oracle Integration Cloud. Email notifications are optional, but by default it is enabled.

    Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

    Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* included in this release about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process:

Artifacts	Value
Integration Process Name	OU Responsys SFTP File Create
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000
Identifier	OUTL-BA-RESP_SFTP_FILE_CREATE
Oracle FTP Adapter Identifier	OUTL-BRT-SFTP_NOTIF_HUB • Operation: Write Files • Output Directory: / • Select a Transfer Mode: Ascii • File Name Pattern: * • Append to Existing File: Unchecked • PGP encryption/description: No PGP Encryption/ Decryption on the file to be sent to external FTP server • Connection Name: OU SFTP for Notif Hub

Artifacts	Value
Oracle Streams Adapter Identifier	OUTL-BRT-STREAM-NOTIF_HUB • Operation: Consume message from stream • Stream Selected: TargetNotificationStream • Partition: Default • Consumer Group: CustomerContactGroup • Max No of records to be fetched: 400 • Specify the option for consuming message: Read Latest • Connection Name: OU OCI Streaming for Notif Hub
OIC Lookup	• OUTL-BRT-NOTIFHUB_ConfigProps • OUTL-BRT-NOTIFHUB_NotifMetaData • OUTL-BRT-NOTIFHUB_Email_ID

OU CCB Notif MetaData Sync (Oracle Integration Cloud Initiated)

This scheduler integration flow synchronizes the Oracle Utilities Customer Care and Billing notification metadata to a Oracle Integration Cloud lookup. The notification types, email template name and isCustomerContacRequired is captured.

For any new environment, always run the integration and sync the NotificationType, EmailTemplateName & isCustomerContactRequired columns in the lookup first before making any changes (manually) to "Priority" and "Email Subject" column. Once the integration syncs the lookup then update the appropriate "Priority" and "Email Subject" details manually in the lookup (dvm).

The following diagram shows a graphical representation of the CCB Notification MetaData sync:
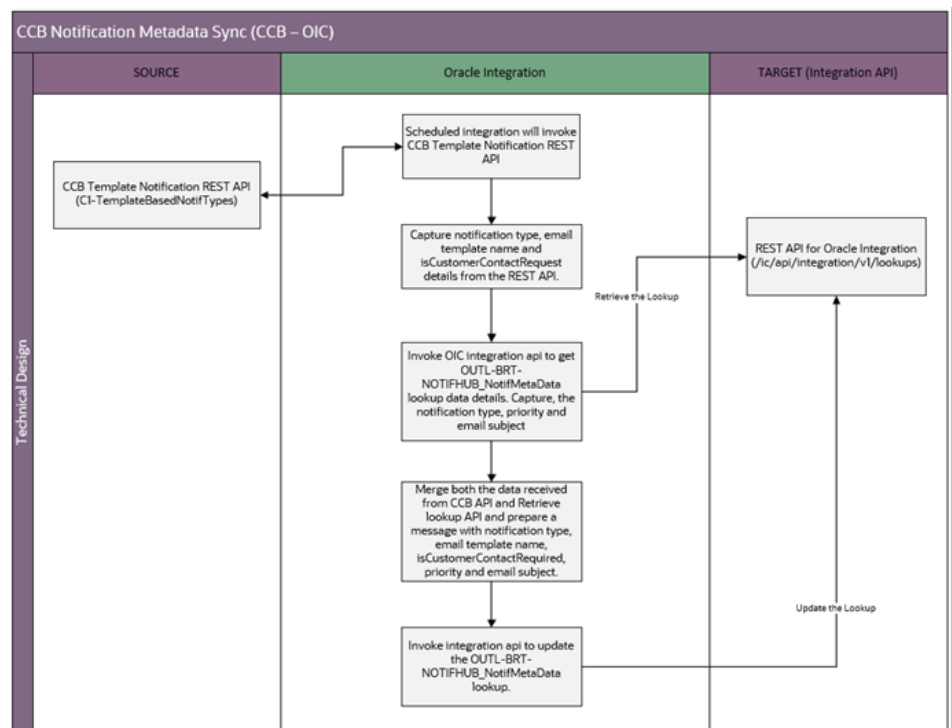
Processing Details

The integration process includes the following activities:

1. This integration process is a scheduleder process and invokes Oracle Utilities Customer Care and Billing Rest API C1-TemplateBasedNotifTypes to capture all the notification. The REST API provides all the notification type with respective email template name and customer's preference to contact or not details in a response collection.

2. This integration then transforms the notification response collection of Oracle Utilities Customer Care and Billing Rest API call and stores the details in OUTL-BRT-NOTIFHUB_NotifMetaData lookup. In this process the notification type, email template name and email template name are populated in DVM. Priority and email subject is not touched in the lookup for until first run of the integration. Subsequent run of this integration depending on the changes in notification type, other data's are updated including priority and email subject.

3. Integration merges both the data that is received from Oracle Utilities Customer Care and Billing and data retrieved from the lookup.

4. Integration updates the lookup with merge data. The final lookup will have notification type, email template name, isCustomerContactRequired, priority and email subject merged details.

5. Integration captures all the notification type (SMS and EMAIL) both in the lookup. For SMS type of notification, template name is not captured in the lookup (DVM).

6. On error, the integration should be re-run.

Refer to Chapter 6: Configuring Lookups, Error Handling, and Email Notifications for error handling information on error scenarios and retry options.

Refer to the *Oracle Utilities Notifications Hub Data Mapping Guide* about mapping information for data elements.

Technical Details

The following table describes the integration processes and the respective artifacts used in this integration process.

Artifacts	Value
Integration Process Name	OU CCB Notif MetaData Sync
Identifier	OUTL-BA-CCB_NOTIF_METADATA_SYNC
Integration Project Name	OUTL-BA-NOTIFICATIONS_HUB.01.24.1000
Oracle Utilities Adapter Connection	OUTL-BRT-REST-CCB_NOTIF_HUB
OIC Lookup	OUTL-BRT-NOTIFHUB_NotifMetaData

Chapter 3

Configuring Oracle Utilities Customer Care and Billing

This chapter elaborates about the configuration of various data, messages and catalog for the integration used by Oracle Utilities Customer Care and Billing. It includes the following sections:

- System Configuration
- Managing Web Service Catalog

System Configuration

To configure the Oracle Utilities Customer Care and Billing setup for the integration:

1. Login to Oracle Utilities Customer Care and Billing.

2. Setup notification types. Refer to the Notification Types section for details.

3. Setup message senders. Refer to the Message Senders section for details.

4. Setup an external system. Refer to the External System section for details.

5. Setup Bill and Collection Event algorithms. Refer to the Bill and Collection Event Algorithms section for more details.

6. Setup feature configuration. Refer to the Feature Configuration section for more details.

7. Setup Notification Preferences Master Configuration and Oracle Utilities Customer Care and Billing Self-Service Integration. Refer to the Master Configuration section for details.

Notification Types

Define the notification types to be used in the integration process between Oracle Utilities Customer Care and Billing and Oracle Utilities Network Management System.

> **Note**: The integration includes starter notification type templates. Additional algorithms are available to support an extended list of outbound notifications from Oracle Utilities Customer Care and Billing. See the Oracle Utilities Customer Care and Billing documentation for reference.

On the **Admin** menu, navigate to the **Notification Type** portal.

> **Note**: For all notification types, a customer contact type can be configured on the notification type to record a record of the notification that includes notification details. To enable, this configure a unique customer contact type per notification type and specify the customer contact type on the notification type.

On the **Admin** menu, navigate to **Add Customer Contact Type**. Add a Customer Contact Type with following details:

- Contact Acton is Record Notification

Add the following Oracle Utilities Network Management System notification types used for this integration:

- Planned Outage Started

- Outage Started

- Planned Outage Restored

- Outage Restored

- Planned Outage Canceled

- Outage Canceled

- Planned Outage Estimated Restoration Time (ERT) Update

- Outage Estimated Restoration Time (ERT) Update

- Planned Outage Scheduled

Add the following Oracle Utilities Customer Care and Billing notification types used for this integration:

- Bill Due Notification

  On the **Admin** menu, navigate to the **Self-Service Task Type** page. Add a **Self-Service Task Type** with following details:

  - Service Task Type Business Object: Bill Due Notification Task Type
  - Related Transaction Object: Bill Due Notification Task
  - Service Task Type Status: Active
  - Service Task Class: Self-Service

  On the **Admin** menu, navigate to the **Notification Type** page. Add a **Notification Type** with following details:

  - Notification Type Business Object: Template-Based Subscription Notification Type
  - Notification Type (Legacy): Bill Due Notification
  - Use Service Task: Enabled
  - Service Task Type: Service Task Type created for Bill Due Notification
  - Template Data Algorithms: Bill Information Template Data
  - Person Contact Types
  - Delivery Types:
    - Delivery Type: Email and/or SMS
    - Template Name: Template Name as created in Notification Delivery System

- Late Payment Notification

  On the **Admin** menu, navigate to the **Self-Service Task Type** page. Add a **Self-Service Task Type** with following details:

  - Service Task Type Business Object: Late Payment Notification Task Type
  - Related Transaction Object: Late Payment Notification Task
  - Service Task Type Status: Active
  - Service Task Class: Self-Service

  On the **Admin** menu, navigate to the **Notification Type** page. Add a **Notification Type** with following details:

  - Notification Type Business Object: Template-Based Subscription Notification Type
  - Notification Type (Legacy): Late Payment Notification
  - Use Service Task: Enabled
  - Service Task Type: Service Task Type created for Late Payment Notification
  - Template Data Algorithms: Bill Information Template Data
  - Person Contact Types

- Delivery Types:
  - Delivery Type: Email and/or SMS
  - Template Name: Template Name as created in Notification Delivery System.

- Invite User Notification

  On the **Admin** menu, navigate to the **Self-Service Task Type** page. Add a **Self-Service Task Type** with following details:

  - Service Task Type Business Object: Web User Invitation Task Type
  - Related Transaction Object: Web User Invitation Task
  - Service Task Type Status: Active
  - Service Task Class: Self-Service

  On the **Admin** menu, navigate to the **Notification Type** page. Add a **Notification Type** with following details:

  - Notification Type Business Object: Template-Based Subscription Notification Type
  - Suppression Criteria: Always Suppressed (C1-ALWY-SUP)
  - Override Delivery Information: Override Guest Delivery Information (C1OVINVGSTDI)
  - Template Data Algorithms: Guest Notification Template Data
  - Person Contact Types
  - Delivery Types:
    - Delivery Type: Email
    - Template Name: Template Name as created in Notification Delivery System.

- Confirm Acceptance Notification

  On the **Admin** menu, navigate to the **Notification Type** page. Add a **DSS-T Notification Type** with following details:

  - Notification Type Business Object: Template-Based Parent Push Notification Type
  - Person Contact Types
  - Delivery Types: Email and/or SMS

  On the **Admin** menu, navigate to the **Notification Type** page. Add a **Notification Type** with following details:

  - Notification Type Business Object: Template-Based Individual Push Notification Type
  - Parent Notification Type: Notification Type created for DSS-T notifications
  - Template Data Algorithms: Guest Notification Template Data
  - Delivery Types:
    - Delivery Type: Email and/or SMS

- Template Name: Template Name as created in Notification Delivery System.

• Collection Notification

On the **Admin** menu, navigate to the **Notification Type** page. Add a parent **Notification Type** with following details:

- Notification Type Business Object: Template-Based Parent Push Notification Type
- Person Contact Types
- Delivery Types: Email and/or SMS

On the **Admin** menu, navigate to the **Notification Type** page. Add a **Notification Type** with following details:

- Notification Type Business Object: Template-Based Individual Push Notification Type
- Parent Notification Type: The Parent Notification Type created above
- Template Data Algorithms: Collection Notification Template Data
- Delivery Types:
  - Delivery Type: Email and/or SMS
  - Template Name: Template Name as created in Notification Delivery System

Notification types can be defined as Template-Based Parent/Individual Push or Template-Based Subscription. Refer to the respective product's online help for more information.

For Template-Based Notification Types, the Template Name defined for each Delivery Types must be a valid campaign in Oracle Responsys.

Template Parameters

Template Data algorithms defined on Notification Types send a set of key/value pairs to the external system. Additional custom Template Data algorithms can be configured on the Notification Type to add more key/value pairs.

Use the following details to configure Notification Templates in Delivery System.

Template parameters sent by Bill Information Template Data algorithm:

- BILL_ID
- BILL_DT
- BILL_DUE_DATE
- AMOUNT_DUE
- CURRENCY_CODE
- CURRENCY_SYMBOL
- MAIN_ACCOUNT_ADDRESS
- SERVICE_ADDRESS

Template parameters sent by Guest Notification Template Data algorithm:

- WEB_USER_NAME
- OWNER_FIRST_NAME
- OWNER_LAST_NAME
- GUEST_FIRST_NAME
- GUEST_LAST_NAME
- GUEST_EMAIL
- GUEST_ACCESS_ROLE
- GUEST_ACCESS_ROLE_DESCR
- MAIN_ACCOUNT_ADDRESS
- SERVICE_ADDRESS

Template parameters sent by Collection Notification Template Data algorithm:

- ARREARS_AMOUNT
- ARREARS_DATE
- CURRENCY_CODE
- CURRENCY_SYMBOL
- MAIN_ACCOUNT_ADDRESS
- SERVICE_ADDRESS

Message Senders

Define a message sender for sending Oracle Utilities Customer Care and Billing template-based notification preferences outbound messages.

On the **Admin** menu, navigate to the **Message Sender** page. Add a message sender for Oracle Utilities Customer Care and Billing template-based notification preferences with the following details:

- Invocation Type: Real Time
- Message Class: RTJSONSNDR (Sender for real-time HTTP / JSON messages)
- MSG Encoding: UTF-8 message encoding
- On the **Context** tab, provide the HTTP URL, Login User, Login Password, HTTP Method and HTTP Timeout to the integration layer.

External System

For Oracle Utilities Customer Care and Billing Notifications, define an external system for Oracle Utilities Customer Care and Billing template-based notification preference messages.

On the **Admin** menu, navigate to the **External System** page. Add a external system for Oracle Utilities Customer Care and Billing template-based notification preference messages.

Add an outbound message type with the following details:

- Outbound Message Type: C1-TMPLNTFRQ (Template Based Notification Request)

- Processing Method: Real-time

- Message Sender: Template-based notification preferences message sender defined above

- Date/Time Format: XSD

- JSON Conversion Method: Base JSON Conversion

Bill and Collection Event Algorithms

Define the Post Bill Completion algorithm.

1. On the **Admin** menu, navigate to the **Customer Class** page.

2. On the **Controls** tab, configure the following algorithm:

- Post Bill Completion: Set Bill Notifications for Self-Service (WX-SSBNOTIFY)

Define the Collection Event algorithm:

1. On the **Admin** menu, navigate to the **Algorithm** page.

2. Add the Collection Event Notification algorithm with the following details:

- Algorithm Type: Collection Event Notification (C1COLLEVTNOT)

- Notification Type: Notification Type defined for Collection Event Notification

3. On the **Admin** menu, navigate to **Collection Event Type** for the notification.

4. Configure the Collection Event Notification algorithm in the **Collection Event Algorithm** field.

Feature Configuration

To setup Invite Guest User and Confirm Acceptance notifications, enable the **Is CX Installed** feature.

1. On the **Admin** menu, navigate to **Feature Configuration**.

2. Add or edit the **General System Configuration** with following options:

- Is CX Installed: Y

Master Configuration

For Oracle Utilities Customer Care and Billing notifications, define the Notification Preferences master configuration:

1. On the **Admin** menu, navigate to the **Master Configuration** portal.

2. Add or edit the **Notification Preferences** master configuration.

3. In the **Default External Communication** section, add the **Default External System** and **Default Outbound Message Type** as defined above for template-

based notification messages. **External System** and **Outbound Message Type** can also be configured separately for each notification type in the **Override External Communication** section.

Define the Oracle Utilities Customer Care and Billing Self-Service Integration master configuration:

1. On the **Admin** menu, navigate to the **Master Configuration** portal.

2. Add or edit the Oracle Utilities Customer Care and Billing Self-Service Integration master configuration. Refer to the product specific online and embedded help for more information on defining each section.

3. In the **Notification Service Tasks** section, add the following notification types:

   • Bill Due Notification: Notification Type defined for Bill Due Notification

   • Late Payment Notification: Notification Type defined for Late Payment Notification

4. In the **Guest User Invitation Notification Types** section, add the following notification types:

   • Invite User: Notification Type defined for Invite Guest Notification

   • Confirm Acceptance: Notification Type defined for Confirm Acceptance Notification

Managing Web Service Catalog

The web service catalog is used by Oracle Integration Cloud to communicate with the respective application. It is used to identify the services that should be retrieved by the Oracle Utilities Adapter. It is configured in Catalog URL in the Oracle Integration Cloud connection.

To configure the catalog in Oracle Utilities Customer Cloud Service:

1. Login to Oracle Utilities Customer Cloud Service.

2. Navigate to the **Web Service Catalog** page either from the **Admin > [W or Integration]** menu or from the **Search** menu.

3. Select **REST Web Service Class**.

4. Add the REST inbound web services mentioned below to the catalog.

Service Type	Service Name	Description
Inbound Web Service	C1-UnsubscribeNotifications	Unsubscribe Notifications
Inbound Web Service	C1-TemplateBasedNotifTypes	Template Based Notification Type
Inbound Web Service	C1-CustomerContactNotification	Create/Update Customer Contact
Inbound Web Service	C1-NotificationPreferencesForAccounts	Notification Preferences for Accounts

5. For outbound messages, add the External System that was set up previously.

For more information about the configuration, refer to the Oracle Utilities Customer Cloud Service documentation.

Chapter 4

Configuring Oracle Utilities Network Management System

The Oracle Utilities Network Management System implementation needs to include Service Alert module. Notifications are generated by the Service Alert service. This chapter focuses on the following:

- NMS Notifications Hub Adapter Configuration

NMS Notifications Hub Adapter Configuration

To configure the NMS Notifications Hub adapter:

1. Configure the Oracle Utilities Network Management System Service Alert command-line option.

   The new command-line option has been added that allows to specify the name of the contact group used to generate outage notifications for Oracle Utilities Notifications Hub.

   ```
   -nhub <group name>
   <group name> - Service Alert contact group name
   ```

2. Configure the Oracle Utilities Notifications Hub Adapter.

   Configuration parameters for the application will be stored in the CES_PARAMETERS database table. The **APP** field contains the 'NHubInterface' value for records related to this application.

   Sensitive configuration parameters (config.auth.pwd and config.auth.user) should be stored in a keystore in production environments. Keystore alias should be the string 'nhubinterface.' followed by the configuration parameter name (example: nhubinterface.config.auth.pwd).

Parameter	Description
config.enabled	Enable/disable the application. Default: true (enabled)
config.auth.url	URL of the OAuth service used to obtain token to be used when accessing Notifications Hub. If not provided, then the Basic HTTP authentication method will be used.
config.auth.scope	The scope value used when requesting OAuth token.
config.auth.user	OAuth client ID or username for Basic HTTP authentication.
config.auth.pwd	OAuth client secret or password for Basic HTTP authentication.
config.publish.url	URL of the Notifications Hub web service.
config.ws.timeout	Timeout in seconds for web service operations. Default: 30 seconds
config.publish.period	Period for publishing outage notifications in seconds. Cannot be set lower than 15 seconds. Default: 60 seconds
config.publish.timeout	Amount of time in minutes after which notifications pending delivery are deleted from the MYC_CSS_PARAM_VIO_TEXT table. Purpose of this configuration option is to avoid old notifications being sent out after Notifications Hub integration has been down for some time. Setting the parameter to 0 disables this functionality. Default: 240 minutes

Parameter	Description
config.publish.nfy_limit	Maximum number of notifications per message (per single web service call). Default: 100
config.publish.size_limit	Maximum size of a single message in bytes. Valid range [100000, 50000000]. 　　**Note**: If a single notification is larger than the message size limit, such notification will not be published. Default: 1000000
config.publish.delete_sent	Delete notifications from the MYC_CSS_PARAM_VIO_TEXT database table once they are successfully published. Else, records are marked as sent but not deleted which can be useful for debugging purposes. Default: true

3. Set up the Oracle Utilities Notifications Hub integration.

    a. Configure and deploy the Oracle Utilities Notifications Hub Adapter.

    b. In Service the Alert Administration tool:

        a. Create a unique contact group for the Oracle Utilities Notifications Hub notification parameters.

        b. Create notification parameters and assign them to the contact group. Only Outage (type 1) notifications are supported. Separate notification parameters are needed for planned and unplanned outages.

        c. Make sure that the 'Customers out' value is set to '1' so that all outages are sent to Oracle Utilities Notifications Hub.

4. In the system.dat file add the '-nhub <contact group>' command-line option to the Service Alert service using the name of the contact group created in step 3.b.a.

A retry from Oracle Utilities Network Management System can be done by changing the MYC_CSS_PARAM_VIO_TEXT.CCS_NOTIFIED value to 'Y'. If MYC_CSS_PARAM_VIO_TEXT.INSERT_TIME is older than 4 hours, the message will not be sent out, but just deleted. Also, INSERT_TIME must be updated to the current time.

For additional information about configuring Oracle Utilities Network Management System services, refer to the **Services Configuration** chapter in the *Oracle Utilities Network Management System Configuration Guide* at: https://docs.oracle.com/en/industries/energy-water/integrations-index.html

Chapter 5

Importing, Configuring, and Testing Integration Connections in Oracle Integration Cloud

This chapter explains the process for importing the project and files needed for the integration and configuration of these connections imported through the project. After a successful import and configuration, the chapter lists out steps to help test the connections. It includes the following sections:

- Importing the Oracle Integration Cloud Project from Oracle Cloud Marketplace
- Verifying the Project Import
- Configuring Connections in Oracle Integration Cloud
- Configuring Agent
- Setting up Certificates for Security

Importing the Oracle Integration Cloud Project from Oracle Cloud Marketplace

All integration points are shipped as part of single project (OUTL-BA.NOTIFICATIONS_HUB.1.24.1000.car) file.

To import a pre-built integration from Oracle Cloud Marketplace:

1. Launch the Oracle Cloud Marketplace portal.

   https://cloudmarketplace.oracle.com/marketplace/en_US/homePage.jspx

2. Click **Applications**.

3. Browse through the list of applications and select the pre-built integration project to import.

   Alternatively, you can search for "Oracle Utilities" and select the **Notifications Hub** integration option.

4. Click **GetApp**.

5. Review and accept the "Oracle Standard Terms and Restrictions".

6. Click **Next**. The My Oracle Support portal opens.

7. Download the integration package from My Oracle Support.

8. When prompted, select the server where the pre-built integration file should be uploaded.

   The pre-built integration is imported as a project file (.car) that is visible on the **Projects** page in Oracle Integration Cloud.

9. On the **Integrations** page, the individual integrations of the imported project file that are designated with a BUILT BY ORACLE message are displayed.

To import a project in Oracle Integration Cloud:

1. Login to Oracle Integration Cloud.

2. Navigate to **Projects**.

3. Click **Add** and then click **Import**.

4. Select the .car file downloaded from Oracle Cloud Marketplace. Drag and drop the file to the browser area.

5. Verify if the project is imported is successfully.

Verifying the Project Import

To verify the project import was successful:

1. Verify whether the following integrations are imported successfully.

   - OU NMS OCI Stream Notif Upload
   - OU OCI Stream Notif linkChannel
   - OU CCB OCI Stream Notif Upload
   - OU OCI Stream Responsys Notif Send
   - OU Responsys CCB Cust Contact Create

- OU Responsys SFTP File Create
- OU Responsys CED Process
- OU CCB Notif MetaData Sync

2. Verify if the following connections are in place.

- OU REST CCB for Notif Hub
- OU OCI Streaming for Notif Hub
- OU REST Responsys for Notif Hub
- OU REST Responsys2 for Notif Hub
- OU REST Responsys3 for Notif Hub
- OU SFTP for Notif Hub

3. Make sure that the following look ups are imported successfully.

- OUTL-BRT-NOTIFHUB_ConfigProps
- OUTL-BRT-NOTIFHUB_Email_ID
- OUTL-BRT-NOTIFHUB_DeliveryChannel
- OUTL-BRT-NOTIFHUB_DeliveryInstance
- OUTL-BRT-NOTIFHUB_DeliveryRouting
- OUTL-BRT-NOTIFHUB_NMS_CCB_NotifTypes
- OUTL-BRT-NOTIFHUB_CustContactStatus
- OUTL-BRT-NOTIFHUB_OriginSystem
- OUTL-BRT-NOTIFHUB_NotifMetaData

4. Make sure that the following library is imported.

- OU Notif Hub Format Datetime

Configuring Connections in Oracle Integration Cloud

After the package is imported and verified, the respective connections have to be configured.

This section describes the procedure to set up the connections.

- Configuring OU REST NMS for Notif Hub Connection
- Configuring OU REST CCB for Notif Hub Connection
- Configuring OU OCI Streaming for Notif Hub Connection
- Configuring OU REST Responsys for Notif Hub Connection
- Configuring OU SFTP for Notif Hub Connection
- Configuring OU REST for Notif Hub Connection

Configuring OU REST NMS for Notif Hub Connection

This connection is used to communicate with Oracle Utilities Network Management System using the Oracle Utilities Adapter.

To configure the OU REST NMS for Notif Hub connection:

1. Add Oracle Utilities Network Management System REST web catalog to the **catalogURL** section.

   The REST Catalog URL follows this format:

   https://{host}:{port}/nms-nhub/catalog.json

2. From the **Security Policy** drop-down list, select the **Basic Authentication or OAuth Client Credentials** policy.

3. Provide **Username** and **Password** for Basic Authentication or provide OAuth details to connect to Oracle Utilities Network Management System.

4. Click **Test** on the upper-right corner.

5. After the connection is tested successfully, click **Save**.

Configuring OU REST CCB for Notif Hub Connection

This connection is used to communicate with Oracle Utilities Customer Cloud Service using the Oracle Utilities Adapter.

To configure the OU REST CCB for Notif Hub connection:

1. Add the Oracle Utilities Customer Care and Billing REST web catalog to the **catalogURL** section.

   The REST Catalog URL follows this format:

   https://{host}:{port}/{tenant}/{domain}/{appName}/rest/openapi/iws/catalog

2. From the **Security Policy** drop-down list, select the **Basic Authentication** policy.

3. Provide **Username** and **Password** to connect to Oracle Utilities Customer Cloud Service.

4. Click **Test** on the upper-right corner.

5. After the connection is tested successfully, click **Save**.

Configuring OU OCI Streaming for Notif Hub Connection

This connection is used to communicate with Oracle Cloud Infrastructure (OCI) Streams using the OCI Streaming adapter.

To configure OU OCI Streaming for Notif Hub connection:

1. Add the Bootstrap Server.

   Know the host and port of the bootstrap server to use to connect to a list of Oracle Cloud Infrastructure Streaming Services.

2. From the **Security Policy** drop-down list, select the **SASL PLAIN Over SSL OSS** policy.

3. Provide the SASL **Username** and **Password**.

4. Provide the **TrustStore**.

5. Provide the **Agent Group**.

6. Click **Test** on the upper-right corner.

7. After the connection is tested successfully, click **Save**.

Refer to https://docs.oracle.com/en/cloud/paas/integration-cloud/stream-service-adapter/create-streaming-service-adapter-connection.html for more information.

Configuring OU REST Responsys for Notif Hub Connection

This connection is used to communicate with Oracle Responsys using the REST adapter.

To configure the OU REST Responsys for Notif Hub connection:

1. Select **Connection Type REST API Base URL**.

2. Add the **Connection URL**.

3. From the **Security Policy** drop-down list, select the **OAuth Custom Two Legged Flow** policy.

4. Provide **Access Token Request**.

5. A sample Access Token Request is as below:

   -X POST https://host/rest/api/v1.3/auth/token -H "Content-Type: application/x-www-form-urlencoded" -d
   "user_name=oic_api@oracleuti&password=1234&auth_type=password"

6. The **user_name** and **password** should be a API user and password.

7. Specify **$access_token as authToken**.

8. Specify access_token_usage as -H Authorization: ${access_token}.

9. Click **Test** at the upper-right corner.

10. After the connection is tested successfully, click **Save**.

> **Note**: The OU OCI Stream Responsys Notif Send integration process invokes three Oracle Responsys instances out of the box. Hence, this Oracle Integration Cloud process requires three Oracle Responsys connections to be configured. Oracle Utilities REST Responsys2 for Notif Hub Connection and Oracle Utilities REST Responsyse3 for Notif Hub Connections can be configured to different Oracle Responsys instances following the same connection configurations.
>
> The Oracle Integration Cloud process validates all the configured connections during activation. If fewer than three different Oracle Responsys instances are used, then same connection configurations should be duplicated in the unused connection to have valid connections in the flow.

Configuring OU SFTP for Notif Hub Connection

This connection is used to communicate with the SFTP Server that Oracle Responsys is connecting to using the Oracle FTP adapter.

To configure OU SFTP for Notif Hub connection:

1. Add the **FTP Server Host Address** and **FTP Server Port**.

2. Know the host and port of the FTP Server to use to connect to get Oracle Responsys related files.

3. From the **Security Policy** drop-down list, select the **SASL PLAIN Over SSL OSS** policy.

4. Provide the **SASL Username** and **Password**.

5. Provide the **TrustStore** and **Agent Group**.

6. Click **Test** on the upper-right corner.

7. After the connection is tested successfully, click **Save**.

Refer to https://docs.oracle.com/en/cloud/paas/integration-cloud/stream-service-adapter/create-streaming-service-adapter-connection.html for more information.

Configuring OU REST for Notif Hub Connection

This connection is used in Oracle Utilities Responsys CCB Cust Contact Create process. It is a trigger connection used to receive requests from OU Responsys CED process.

To configure the OU REST for Notif Hub connection:

1. Select **Security** to be the **Basic** authentication.

2. Click **Test** on the upper-right corner.

3. After the connection is tested successfully, click **Save**.

Configuring Agent

Create an agent group in Oracle Integration Cloud and install agent on the on-premises server before creating/activating an integration in which messages are exchanged between the on-premises applications (example: Oracle Utilities Network Management System, Oracle Utilities Customer to Meter) and Oracle Integration Cloud. The agent related configurations are needed only if the server points to an on-premises application and to access OCI Streaming.

> **Note**: A direct connection from Oracle Integration to the private endpoint OCI Streaming Service does not work. You must install the connectivity agent on the same network (a virtual machine (VM) created on the same virtual cloud network (VCN) and subnet).

This section includes:

- Creating an Agent Group
- Downloading Agent Installer
- Installing On-Premises Agent

Creating an Agent Group

Create an agent group in Oracle Integration Cloud before running the agent installer. When the on-premises agent is installed in the environment, the on-premises agent is associated with the agent group identifier.

For a single Oracle Integration Cloud instance, you can create up to five agent groups. Creating the agent group also creates the necessary queues required for message exchange.

To create an agent group:

1. Login to Oracle Integration Cloud.

2. On the **Home** page, click **Agents**.

3. Click **Create Agent Group**.

4. Enter the following information:

   • Agent Group Name

   • Identifier

     Note: The agent group name and identifier must be same.

   • Agent Type: "Connectivity Agent"

   • Description

5. Click **Create**.

Downloading Agent Installer

Download the agent installer from Oracle Integration Cloud and run the installer to install the on-premises agent in your local environment. During the installation, associate the agent with the Agent Group Identifier generated when creating an agent group in Oracle Integration Cloud.

For more information on agent installer, see https://docs.oracle.com/en/cloud/paas/integration-cloud/integrations-user/downloading-and-running-premises-agent-installer.html.

Installing On-Premises Agent

To install an on-premises agent:

1. Login to Oracle Integration Cloud.

2. On the **Home** page, click **Agents**.

3. Click **Download**.

4. Select **Connectivity Agent**.

5. Select **Save File** when prompted to save the file to a directory location on your on-on-premises host.

6. Navigate to that directory and unzip **oic_connectivity_agent.zip**.

7. Change the file permissions to be executable.

8. Download the Oracle Utilities Customer Care and Billing certificate. Upload it by running the following command from the agent home directory.

```
keytool -import -file directoryPath/sample.crt -alias SampleCert -keystore <Agent_Home>/agenthome/agent/cert/keystore.jks
```

or

```
keytool -import -file directoryPath/sample.crt -alias SampleCert -keystore <Agent_Home>/agenthome/agent/cert/keystore.p12
```

9. Modify **InstallerProfile.cfg** to include the following information.

a. Provide the oic_URL value with the OIC SSL host name.

Example: https://OIC_host:OIC_port

b. Provide the agent_GROUP_IDENTIFIER. It should be the agent group created in Oracle Integration Cloud.

c. Set the proxy parameters if the connectivity agent is used with a proxy in the on-premises environment.

d. Set the JAVA_HOME property to the directory/folder where JDK is installed.

Note: Before running the connectivity agent installer, perform the steps listed on this page: https://docs.oracle.com/en/cloud/paas/integration-cloud/utilities-adapters/you-begin-setting-oracle-utilities-adapter.html#GUID-7F770AD1-5B87-4C62-968A-3AB30D043835

e. Run the connectivity agent installer from the command prompt.

```
java -jar connectivityagent.jar
```

f. Provide the Oracle Integration Cloud credentials when prompted.

g. Wait for a successful installation message to appear.

After the installation is complete, an agent instance is created to interact with Oracle Integration Cloud.

To verify if the agent instance was created:

1. Navigate to the **Agents** page in Oracle Integration Cloud.

2. Check if the agent count for your **Agent Group** is increased by one.

3. Click the number to view the agent details.

For more details, refer to Oracle Integration Cloud documentation at: https://docs.oracle.com/en/cloud/paas/integration-cloud-service/index.html

Setting up Certificates for Security

Important! Please skip this section if there are valid CA certificates for the integration.

If there are no valid certificates for this integration, download the Oracle Utilities Customer Care and Billing certificates or Oracle Utilities Network Management System application and upload them to Oracle Integration Cloud to handshake with Oracle Utilities Customer Care and Billing.

To download the Oracle Utilities Customer Care and Billing certificate:

1. Login to Oracle Utilities Customer Care and Billing.

2. Click the URL on the top-left corner.

3. On the **Security** tab, click **View Certificate**.

4. On the **Details** tab, click **Export**.

5. Save the certificate.

To upload the certificate to Oracle Integration Cloud:

1. Login to Oracle Integration Cloud with Admin credentials.

2. Navigate to **Settings** > **Certificates**.

3. On the **Certificate** window, click **Upload**.

4. Select **Certificate Type** as **Trust Certificate**.

5. Provide the **Certificate Alias Name**.

6. Select the certificate to upload.

7. Click **Upload**.

Chapter 6

Configuring Lookups, Error Handling, and Email Notifications

This chapter focuses on the lookups configuration, handling business and technical errors, and sending email notifications in this integration. It includes the following sections:

- Configuring Lookups
- Configuration Properties
- Error Handling
- Email Notifications

Configuring Lookups

The following table lists the lookups and integration processes where these lookups are used.

Lookup Name	Integration Name	Purpose
OUTL-BRT-NOTIFHUB_NotifMetaData	Oracle Utilities Responsys CCB Cust Contact Create	Maps the template to the email subject (EmailTemplateName)
OUTL-BRT-NOTIFHUB_NotifMetaData	Oracle Utilities CCB Notif MetaData Sync	Maps the lookup columns NotificationType,EmailTemplateName,CustomerContactRequired to the CCB Notif MetaData API.
OUTL-BRT-NOTIFHUB_NotifMetaData	Oracle Utilities OCI Stream Responsys Notif Send	Maps the CCB Notification Types to a priority, 'H' is used for High Priority.
OUTL-BRT-NOTIFHUB_DeliveryChannel	Oracle Utilities OCI Stream Responsys Notif Send	Maps CCB Delivery Type to the Integration-defined delivery type of "sms" or "email".
OUTL- BRT-NOTIFHUB_DeliveryInstance	Oracle Utilities OCI Stream Responsys Notif Send	Maps the Integration-defined Oracle Responsys delivery instance ID to the actual Oracle Responsys identifier. The ResponsysId can be anything that identifies an Oracle Responsys instance. Example: Description, URL, Instance ID, etc. This Responsys ID is used in the error email notifications sent to identify the Oracle Responsys instance where the invocation failed. **Note**: The values in the Intg_DeliveryInstanceId should be RESP1, RESP2, RESP3.

Lookup Name	Integration Name	Purpose
OUTL-BRT-NOTIFHUB_DeliveryRouting	• Oracle Utilities OCI Stream Notif linkChannel • Oracle Utilities CCB OCI Stream Notif Upload	Maps CCB CIS Division to the Oracle Responsys delivery Instance to route the message. The Oracle Responsys instances provided out of the box are identified as RESP1, RESP2 and RESP3 in the DeliveryInstance column. **Note**: These values in the DeliveryInstance column should match Intg_DeliveryInstanceId of OUTL- BRT- NOTIFHUB_ DeliveryInstance.
OUTL-BRT-NOTIFHUB_ConfigProps	• Oracle Utilities OCI Stream Responsys Notif Send • Oracle Utilities OCI Stream Notif linkChannel • Oracle Utilities CCB OCI Stream Notif Upload • Oracle Utilities NMS OCI Stream Notif Upload	Maps the PropertyName column to the respective Value column. Do not change the value in the PropertyName column.
OUTL-BRT-NOTIFHUB_Email_ID	• Oracle Utilities OCI Stream Notif linkChannel • Oracle Utilities CCB OCI Stream Notif Upload • Oracle Utilities NMS OCI Stream Notif Upload • Oracle Utilities OCI Stream Responsys Notif Send	Maps the "from" and "to" recipient values to the mentioned email IDs. Configure multiple emails using a comma to separate the email IDs.

Lookup Name	Integration Name	Purpose
OUTL-BRT-NOTIFHUB_NMS_CCB_NotifTypes		Maps the CCB Notification Types to the NMS Trigger Code and Planned/Unplanned Flag values. The DVM column, TriggerCode_Planned is combination of two parts. The first part is a number which represents NMS trigger code and the second is the planned or unplanned flag Y or N. "N" means unplanned outage and "Y" mean Planned outages. For example: "1N" means Unplanned outage started and "4N" means Unplanned outage cancelled. The flags are case sensitive and must be in uppercase. The following are the NMS trigger codes and can be combined with Planned or Unplanned outages flags: 1 - Outage started 2 - Outage restored 3 - Outage scheduled 4 - Outage scheduled 5 - Outage updated ERT **Note**: "Unplanned outage schedule (3N) is not supported".
OUTL-BRT-NOTIFHUB_CustContactStatus	Oracle Utilities Responsys CCB Cust Contact Create	Maps the customer contact status between Oracle Responsys status and CCB create customer contact status.
OUTL-BRT-NOTIFHUB_OriginSystem	Oracle Utilities Responsys CCB Cust Contact Create	Originating system and its code details.

Editing Lookups

To edit a lookup to add or update any value in it:

1. Log in to Oracle Integration for Cloud.

2. Navigate to **Integrations** > **Designer** > **Lookups**.

3. Click the look up to edit.

4. Make the necessary changes.

5. Click **Save** and **Close**.

> **Note**: While editing a lookup, if the changes are not reflected during the runtime, make sure to deactivate and activate the integration that is using the edited DVM for the changes to reflect. Refer to the Activating Integration Flows section for more details.

Configuration Properties

OUTL-BRT-NOTIFHUB_ConfigProps.dvm contains the properties that can be defaulted in the integration. It also contains a flag to enable email notifications.

Property Name	Sample Value	Description	Used in Integration Process Name
nms.datafield.propertylist	OUTAGETIME,CUSTOMERSOUT,CAUSE, ESTRESTTIME, CREWETA, COMPLETIONTIME		Oracle Utilities NMS OCI Stream Notif Upload
email.flag	True	If this value is set to true, an email notification will be sent out. Valid values: true/false Default: true	• Oracle Utilities OCI Stream Responsys Notif Send • Oracle Utilities NMS OCI Stream Notif Upload • Oracle Utilities OCI Stream Notif linkChannel • Oracle Utilities OCI Stream Notif Upload
resp.email.permission	OPTOUT	Merges the rule corresponding to "defaultPermissionStatus" field in Oracle Responsys API. Do not change.	Oracle Utilities OCI Stream Responsys Notif Send
resp.reject.channel.empty	E,M	Merges the rule corresponding to "rejectRecordIfChannel Emtpy"field in Oracle Responsys. Do not change.	Oracle Utilities OCI Stream Responsys Notif Send
resp.optout.value	O	Merges the rule corresponding to "optoutValue" field in Oracle Responsys.	Oracle Utilities OCI Stream Responsys Notif Send
resp.optin.value	I	Merges the rule corresponding to "optinValue"field in Oracle Responsys.	Oracle Utilities OCI Stream Responsys Notif Send
resp.update.on.match	REPLACE_ALL	Merges the rule corresponding to "updateOnMatch" field in Oracle Responsys. Controls how the existing record should be updated.	Oracle Utilities OCI Stream Responsys Notif Send

Property Name	Sample Value	Description	Used in Integration Process Name
resp.match.operator	NONE	Merges the rule corresponding to "matchOperator" field in Oracle Responsys. Operator to join the match column names.	Oracle Utilities OCI Stream Responsys Notif Send
Form.base64.encoded	true	Indicates if the Unsubscribe data in the files is base64encoded.	Oracle Utilities Responsys CCB Pref Unsubscribe
max.error.count	30	Maximum count of errors at which to send an email notification.	Oracle Utilities Responsys CCB Pref Unsubscribe Oracle Utilities Responsys CCB CCB Contact Create

Error Handling

This section provides information on how to handle integration errors, and how to resubmit the instances after rectification.

- Error Handling Procedures
- Summary of Integration Error Handling
- Resubmitting the Error Instances in Oracle Integration Cloud

Error Handling Procedures

For this integration, the errors are handled in Oracle Integration Cloud through the following procedures:

- Synchronous Flow Error Handling
- Asynchronous Flow Error Handling

Synchronous Flow Error Handling

This error handler sends back the errors to the corresponding source system in the same flow.

Technical Fault

This fault occurs in case of data mismatch in DVM, or any Xpath-related error. When this error occurs, the flow immediately turns to the global fault handler, and the fault is sent back to the respective system.

Remote Fault

This fault occurs when there is a connectivity issue. When this occurs, the flow immediately turns to the global fault handler, and the fault is sent back to the respective system.

Business Fault

This fault occurs in case of an invalid data error in the target system. When this occurs, the fault is thrown to the respective source system.

Asynchronous Flow Error Handling

Technical Fault

This fault occurs in case of data mismatch in DVM, or any Xpath-related error. When this error occurs, the flow immediately turns to the global fault handler, and an optional email can be sent to the user.

Remote Fault

This fault occurs when the target system is down. When this error occurs, the flow immediately turns to the global fault handler, and an optional email can be sent to the user.

Business Fault

This fault occurs in case of an invalid data business fault in the target system. When this occurs, the flow immediately turns to the global fault handler, and an optional email can sent to the user.

Summary of Integration Error Handling

Integration Process: OU OCI Stream linkChannel

Type of Error	Action	Notification Type	Retry
Source Stream is down	The process is not invoked.	No Notification	Once Source Stream is up, the poller picks up the messages on the stream run.
Technical or Remote (CCB or Target Stream is unreachable)	Message is published to Source Stream (Rolling back the messages)	Email (optional)	Poller picks up the messages on the next run. **Note**: Inactivate this integration process until all applications (CCB, Stream) are up to avoid the continuous reprocessing of messages that would fail.

Integration Process: OU NMS OCI Stream Notif Upl

Type of Error	Action	Notification Type	Retry
Technical or Remote (Source OCI Stream is unreachable)	The only error scenarios expected are technical errors while accessing OCI Stream, such as: password expired, certificates expired. The expectation of OCI Stream call is rare programmed schedule must be paused.	Email	After fixing OCI Stream connectivity from NMS, resend the notification.

Integration Process: OU Responsys CCB Cust Contact Create

Type of Error	Action	Notification Type	Retry
400 or 500 error status when CCB Create Customer Contact API is invoked	As part of the CED/Pending file processing, in the message collection, each errored record or message is captured. After reaching the max error count, an email with the details of the failed records is sent.	Email is sent with error details for each CED/Pending file type. Also, in the global fault handler, an error email notification is sent if the notification flag is turned on.	Reprocessing happens on the next schedule job process.
Not 400 or 500 error status when CCB Create Customer Contact API is invoked but other status code except 200. Example: 403, 401 error code	The integration process will be stopped, and no further messages will be processed from the file.	In global error handler scope, an email notification will be sent if notify error flag is turn 'on'.	To retry (example: 401) passwords expired, does not make sense, hence should not be retried in this scenario. After fixing the expired password, the flow can be scheduled again.
Oracle Responsys SFTP server being down/ unreachable	The integration will not pick any files for further processing.	In global error handler scope, an email notification will be sent if notify error flag is turn 'on'.	Reschedule the parent process in integration process (Oracle Utilities Responsys CED Process).

Integration Process: OU Responsys CED Process

Type of Error	Action	Notification Type	Retry
Oracle Utilities Responsys CCB Cust Contact Create integration service is unavailable	API call will be failed and CED file will be processed. The process is terminated.	In global error handler scope, an email notification will be sent if notify error flag is turn on.	Reschedule the integration process.

Integration Process: OU Responsys SFTP File Create

Type of Error	Action	Notification Type	Retry
Responsys SFTP server being down/unreachable.	The integration will put back the consumed message in OCI stream.	In global error handler scope, an email notification will be sent if notify error flag is turn on.	OCI message will be consumed automatically on next schedule run.
OCI stream down/ unreachable	Email is sent.	In global error handler scope, an email notification will be sent if notify error flag is turn on.	OCI message will be consumed automatically on next schedule run.

Integration Process: OU OCI Stream Responsys Notif Send

Type of Error	Action	Notification Type	Retry
Not 401 or 500 error status when Responsys API is invoked	Error messages from each invoke in a batch for a particular Oracle Responsys instance, along with account ID, template name and delivery channel is collected in a collection.	A separate email is sent for errors in each Oracle Responsys instance.	These notification messages cannot be recovered.
Error status that is 500 or 401 when Responsys API is invoked	Error messages from each invoke in a batch for a particular Oracle Responsys instance, along with account ID, template name and delivery channel is collected in a collection.	A separate email is sent with error collection of each Responsys instance.	These notification messages are republished back to TargetNotificationStream, until the issue is resolved. **Note**: If the resolution takes a while, inactivate the process to avoid receiving emails and republishing.
Technical or Remote (TargetNotificationStream OCI Stream is unreachable)	During polling, if the OCI Stream is unreachable, the OIC instance is not instantiated.	Email (optional)	Poller will pick up the messages on next try.
Invalid delivery channel or Oracle Responsys instance	Error messages from each invoke in a batch for a particular Oracle Responsys instance, along with account ID, template name and delivery channel is collected in a collection.	An email is sent with the error collection for each batch.	These notification messages cannot be recovered.

Integration Process: OU CCB OCI Stream Notif Upload

Type of Error	Action	Notification Type	Retry
Technical or Remote (Target Stream is unreachable)	• Fault is logged in the OIC Instance. • Fault is rethrown to Global handler.	No Notification	CCB must resend the notification outbound message to OIC.
Any Global	Fault is rethrown back to CCB.	Email (Optional)	CCB must resend the notification outbound message to OIC.

Integration Process: OU Responsys CCB Pref Unsubscribe

Type of Error	Action	Notification Type	Retry
Business Error when processing records in file	The errors are collected in an error collection and the processing continues.	An email notification is sent when error count reaches the max.error.count defined in the configuration lookup.	Based on the errors in the error notification email any updates to the record is done manually.
Technical or Remote (CCB or SFTP location is unreachable)	Global fault is thrown	Email (optional)	Reschedule the process once the issue is resolved.

Integration Process: OU CCB Notif MetaData Sync

Type of Error	Action	Notification Type	Retry
Source Stream (CCB) is down		Email notification (if configured)	On Error, run the integration again.
Technical or Remote fault		Email notification (if configured)	On Error, run the integration again.
OCI Rest call to create/modify DVM api		Email notification (if configured)	On Error, run the integration again.

Resubmitting the Error Instances in Oracle Integration Cloud

The resubmit option is only available for asynchronous flows, as are all the flows initiated by Oracle Utilities Network Management System for this integration.

To resubmit the error instances in Oracle Integration Cloud:

1. Log in to Oracle Integration Cloud.

2. Navigate to **Integrations** > **Monitoring** > **Errors**.

3. Select the integration to resubmit.

4. Click the **Resubmit** icon.

Note: Retry or Resubmission error instance from the **OIC Monitoring** page may not be applicable for the OU NMS OCI Stream Notif Upload integration flow. Refer to the

Summary of Integration Error Handling section on how to retry/resubmitted errored transactions/record.

Email Notifications

This pre-built integration includes a configurable email notification.

To receive an email notification:

1. Login to Oracle Integration Cloud.

2. Navigate to **Integrations** > **Designer** > **Lookups**.

3. Edit the **OUTL-BRT-NOTIFHUB_ConfigProps** lookup.

   a. Change the **send.email.flag** property value to 'true'.

4. Edit the **OUTL-BRT-NOTIFHUB** lookup.

   a. In the **from** field, enter the email ID to receive an email from.

   b. In the **to** field, enter the email ID to send the email to.

   c. In the **Email_Id** field, provide the comma separated email IDs.

      Note: In the **OUTL-BRT- NOTIFHUB_Email_ID** lookup, do not edit the values provided under the **Recipient** column.

Chapter 7

Customizations

This chapter describes options for customizing the integration to meet specific business requirements.

In Oracle Integration Generation 3, you can extend (customize) an integration in an accelerator project by adding and configuring an extension group. An extension group enables you to extend your integration by adding invoke connections; stitch, for-each, switch, map, and integration actions; and global variables to the integrations in your accelerator project.

For more details about how to perform these changes, refer to the Manage a Project section oin the **Using Integrations in Oracle Integration 3** chapter at: https://docs.oracle.com/en/cloud/paas/application-integration/integrations-user/manage-project.html#GUID-BD7207BB-19D8-42D1-8D9E-7FC4D50E1B57

In addition, a knowledge base article is available at: https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=407954934694303&id=3017378.1&_adf.ctrl-state=611abf54g_77

Chapter 8

Activating and Testing the Integration Flows

This chapter provides an overview of how integration flows are activated and tested. It includes the following sections:

- Prerequisites
- Activating Integration Flows
- Testing the Integration Flows

Prerequisites

Make sure the catalog in Oracle Utilities Customer Care and Billing is configured completely to activate an integration process.

Activating Integration Flows

To activate the integration flows:

1. Navigate to the integration to activate.

2. Drag the slider for that integration. When prompted to enable tracing, click **Yes** to view the instances.

3. Click **Activate**.

The integration takes time to get activated. The activated integration appears at the top of the integrations list.

Testing the Integration Flows

The following table lists the integration end point URLs and the respective applications in which these endpoints need to be configured. Configure the same and perform end-to-end testing.

Integration Name	End Point URL to be Configured	Application to be Configured
OU NMS OCI Stream Notif Upload	https:// OIC_instance.REGION.ocp.oraclecloud.com:OIC_ Port/ic/api/integration/v2/flows/oracleutilities/ project/OUTL-BA-NOTIFICATIONS_HUB/ OUTL-BA-NMS_STREAM_NOTIF_UPL/1.0/ NMS/Outages	Oracle Utilities Network Management System
OU CCB OCI Stream Notif Upload	https:// OIC_instance.REGION.ocp.oraclecloud.com OIC_Host:OIC_Port/ic/api/integration/v2/flows/ oracleutilities/project/OUTL-BA-NOTIFICATIONS_HUB/OUTL-BA-CCB_STREAM_NOTIF_UPLOAD/1.0/ notifications **Note**: If the base flow is cloned for customization and the customized flow is used, replace OUTL-BA- CCB_STREAM_NOTIF_UPLOAD in the URL with identifier of the cloned flow.	Oracle Utilities Customer Care and Billing

Chapter 9

Monitoring and Troubleshooting

This chapter provides information about monitoring and troubleshooting the integration. It includes the following sections:

- Oracle Utilities Customer Care and Billing
- Oracle Integration Cloud

Oracle Utilities Customer Care and Billing

This section provides information about monitoring Oracle Utilities Customer Care and Billing.

Oracle Utilities Customer Care and Billing Error Logs

Monitoring the error logs is possible only in on-premises applications. Applications on cloud cannot access the error logs.

The following error logs can be monitored for Oracle Utilities Customer Care and Billing:

- Errors related to the online integration invocation from Oracle Utilities Customer Care and Billing are stored in the CCB_ENVIRONMENT_NAME/ logs/system folder.

  Example: V27_CCB_ORA_WLS/logs/system\

For more information about errors and notifications, see the Oracle Utilities Customer Care and Billing documentation.

Oracle Integration Cloud

This section focuses on the monitoring Oracle Integration Cloud and troubleshooting any issues that occur during the integration activation.

Monitoring Integration Flows

Integration flows are monitored using the following:

- Dashboard
- Cloud Logs

To monitor the integration flows from the **Oracle Integration Cloud dashboard**:

1. Login to Oracle Integration Cloud.

2. On the **Home** page, click **Observability**.

3. Select any of the following as required:

   - **Dashboards**: To monitor the complete dashboard of integration.

   - **Integrations**: To monitor each integration.

   - **Instances**: To monitor instance and flow trace/activity stream of the integration.

   - **Error**: To monitor the integrations in 'error' state. Re-submit the asynchronous integration flows.

To monitor the integration flows using **Oracle Integration Cloud logs**:

1. Login to Oracle Integration Cloud.

2. On the **Home** page, click **Monitoring**.

3. On the navigation pane, click **Dashboards** to view the overall success/failure rate of the integration.

4. Navigate to the **Logs** menu.

5. In the right pane, click the link to show options for downloading the Oracle Integration Cloud logs or diagnostics logs.

6. In case of any issues, attach the diagnostic logs to a service request for help.

Troubleshooting

If an activation fails, the **Integrations** page displays an error message.

To troubleshoot the activation error:

1. Click **Download Diagnostic Logs** to download the logs for diagnosing the issue.

2. Select **Enable Tracing**.

   TRACE ENABLED is displayed next to ACTIVE.

Some of the sample cases are as follows:

- For any connectivity errors while activating the integration, make sure the trigger connection is successful. Test the connection and refresh the metadata, and then activate the integration.

- If the integration (Oracle Utilities Customer Care and Billing initiated flows) is activated for the first time, ensure the Oracle Utilities Customer Care and Billing catalog is configured accurately.

- If AM_VIO_TEXT.INSERT_TIME is older than 4 hours, the message will not be sent out but just deleted. Also, INSERT_TIME must be updated to the current time.

Chapter 10

Sample Messages

This chapter includes the following sample messages:

- Sample Message Flow from NMS through Link Notification and Channel Information Process
- Sample CCB Notification Message to the TargetNotificationStream OCI Stream
- Sample Outage NMS Notification Message to SourceNotificationStream OCI Stream Message
- Sample OU CCB Notif MetaData Sync Message

Sample Message Flow from NMS through Link Notification and Channel Information Process

Message Consumed from Source Notification Stream	Message To and From Oracle Utilities Customer Care and Billing	Message Published to Target Stream

Message Consumed from Source Notification Stream

```
{
"accountId": "123456789",
"notificationType": "OUTAGE",
"messageId": "111",
"origin": "NMS",
"notificationTemplate": "",
"data": [
    {
    "key": "REASON",
    "value": "STORM"
    },
    {
    "key": "ERT",
    "value": "12-10-
2021T10:00AM"
    }
]
}
```

Message To and From Oracle Utilities Customer Care and Billing

Input:

```
{
"C1AccNtfPref": {
    "accounts": [
    {
    "sequence": "10",
    "accountId": "123456789",
    "notificationType": " OUTAGE"
    }
    ]
}
}
```

Output:

```
{
"C1AccNtfPref": {
    "accounts": [
    {
    "sequence": "10",
    "accountId": "123456789",
    "notificationType": "OUTAGE",
    "notificationInstanceId":"000000000
01",
    "cisDivision": "OH",
    "notificationPreferences": [
    {
    "notificationPreferenceId":
"18384304332111",
    "customerContactId":"",
    "subscriptionOrPush":"",
    "isOptOutAllowed":"",
    "isCustomerContactRequired":"",
    "deliveryType": "EMAIL",
    "personContact": {
    "personId": "PerId123",
    "contactId": "Cont111",
    "language": "ENG",
    "firstName": "ABC Business",
    "lastName": "",
    "contactValue": "abcbus@email.com",
    "template": "OUTEMAIL"
    }
    },
    {
    "notificationPreferenceId":
"18586367394027","customerContactId
":"",
    "subscriptionOrPush":"",
    "isOptOutAllowed":"",
    "isCustomerContactRequired":"",
    "deliveryType": "SMS",
    "personContact": {
    "personId": "PerId123",
    "contactId": "Cont333",
    "language": "ENG",
    "firstName": "Ben",
    "lastName": "Baker",
    "contactValue": "19734445678",
    "template": "OUTSMS"
    }
    }
    ]
    }
    ]
}
}
```

Message Published to Target Stream

Two messages are published in this sample.
Message 1

```
{
    "messageId":
"45303230363835333838363231373
3738 ",
    "origin": "NMS",
    "accountId": "123456789",
    "notificationType": "OUTAGE",
    "cisDivision": "OH",
    "contactId": "Cont111",
    "personId": "PerId123",
    "language": "ENG",
    "firstName": "ABC Business",
    "lastName": "",
    "deliveryChannel": "EMAIL",
    "deliveryValue":
"abcbusemail.com",
    "notificationTemplate":
"OUTEMAIL",
    "deliveryInstance": "RESP1",
    "CCId":"1234567890",
    "notifPreferenceId":"04878213
560769",
    "optOutAllowed":"Yes",
    "custContactNeeded":"Yes",
    "subscriptionType":"Subscript
ion",
    "notifEventInstanceId":"00000
000001",
    "sourceMessageId":"111",
    "data": [
        {
        "key": "REASON",
        "value": "STORM"
        },
        {
        "key": "ERT",
        "value": "12-10-
2021T10:00AM"
        }
    ]
}
```

Message 2

```
{
    "messageId":"5703032303638353
33838363231373738 ",
    "origin": "NMS",
    "accountId": "123456789",
    "notificationType": "OUTAGE",
    "cisDivision": "OH",
    "contactId": "Cont333",
    "personId": "PerId123",
    "language": "ENG",
    "firstName": "Ben",
    "lastName": "Banter",
    "deliveryChannel": "SMS",
    "deliveryValue":
"19734445678",
    "notificationTemplate":
"OUTSMS",
    "deliveryInstance": "RESP1",
    "CCId":"",
```

Message Consumed from Source Notification Stream	Message To and From Oracle Utilities Customer Care and Billing	Message Published to Target Stream

		`"notifPreferenceId":"18586367`
		`394027",`
		`"optOutAllowed":"",`
		`"custContactNeeded":"",`
		`"subscriptionType":"",`
		`"notifEventInstanceId":"00000`
		`000001",`
		`"sourceMessageId":"111",`
		`"data": [`
		`    {`
		`        "key": "REASON",`
		`        "value": "STORM"`
		`    },`
		`    {`
		`        "key": "ERT",`
		`        "value": "12-10-`
		`        2021T10:00AM"`
		`    }`
		`]`
		`}`

Sample CCB Notification Message to the TargetNotificationStream OCI Stream

CCB Outbound Message	Message Published to TargetNotificationStream

```
{
    "sendDetail": {
        "accountId": "4128974892",
        "cisDivision": "CA",
        "notificationType": "BILLDUE",
        "notificationInstanceId":"00000000001",
        "syncRequestId": null,
        "timezone" :"USEstern",
        "deliveryInfo": [
            {
                "deliveryType": "SMS",
                "contactValue": "10000000000",
                "templateName": "BILLDUE-SMS",
                "dndStartTime": "",
                "dndEndTime": "",
                "firstName": "John",
                "lastName": "Doe",
                "language": "ENG",
                "personId": "9251189340",
                "contactId": "9252198992",
                "overrrideDND": "",
                "notificationPreferenceId":
                "04878213560769",
                "customerContactId":"1234567890",
                "subscriptionOrPush":"C1SU",
                "isOptOutAllowed":"true",
                "isCustomerContactRequired":"true"
            }
        ],
        "templateParm": [
            {
                "key": "AMOUNT_DUE",
                "value": "6080.05"
            },
            {
                "key": "BILL_DT",
                "value": "2022-01-01"
            },
            {
                "key": "BILL_DUE_DATE",
                "value": "2022-01-17"
            },
            {
                "key": "BILL_ID",
                "value": "412219634128"
            },
            {
                "key": "CURRENCY_CODE",
                "value": "USD"
            },
            {
                "key": "CURRENCY_SYMBOL",
                "value": "$"
            },
            {
                "key": "MAIN_ACCOUNT_ADDRESS",
                "value": " Street, San Francisco,
                CA,94104, USA, Single family
                home,6206198576"
            },
            {
                "key": "SERVICE_ADDRESS",
                "value": " Street, San Francisco,
                CA,94104, USA, Single family
                home,6206198576"
            }
        ]
    }
}
```

```
{
    "messageId": "353032303638353338383632231373738
    "null,
    "origin": "CCB",
    "accountId": "4128974892",
    "notificationType": "BILLDUE",
    "cisDivision": "CA",
    "contactId": "9252198992",
    "personId": "9251189340",
    "language": "ENG",
    "firstName": "John",
    "lastName": "Doe",
    "deliveryChannel": "SMS",
    "deliveryValue": "10000000000",
    "notificationTemplate": "BILLDUE-SMS",
    "deliveryInstance": "RESP1",
    "CCId":"1234567890",
    "notifPreferenceId":"04878213560769",
    "optOutAllowed":"Yes",
    "custContactNeeded":"Yes",
    "subscriptionType":"Subscription",
    "notifEventInstanceId":"00000000001",
    "sourceMessageId":"",
    "timezone":"EST"
    "eventCapturedDTTM":"15-07-2022 19:29:59"
    "data": [
        {
            "key": "AMOUNT_DUE",
            "value": "6080.05"
        },
        {
            "key": "BILL_DT",
            "value": "2022-01-01"
        },
        {
            "key": "BILL_DUE_DATE",
            "value": "2022-01-17"
        },
        {
            "key": "BILL_ID",
            "value": "412219634128"
        },
        {
            "key": "CURRENCY_CODE",
            "value": "USD"
        },
        {
            "key": "CURRENCY_SYMBOL",
            "value": "$"
        },
        {
            "key": "MAIN_ACCOUNT_ADDRESS",
            "value": " Street, San Francisco, CA, 94104,
            USA, Single family home, 6206198576"
        },
        {
            "key": "SERVICE_ADDRESS",
            "value": " Street, San Francisco, CA, 94104,
            USA, Single family home, 6206198576"
        }
    ]
}
```

Sample Outage NMS Notification Message to SourceNotificationStream OCI Stream Message

NMS Notification Message (JSON)	SourceNotificationStream (JSON)

```
{
"notifications": [
    {
    "id": 9005,
    "trigger": 2,
    "planned": "N",
    "event": {
    "eventidx": 6752,
    "outagetime": 1707409180,
    "completiontime": 1707409334,
    "estresttime": 1707412861,
    "customersout": 5,
    "cause": "TROUBLE_CALL",
    "creweta": 0
    },
    "customers": [
    {
    "acct": "2200639",
    "addr": "13959 GREENLAND AVE NW, Lake Twp., OH, 44685"
    },
    {
    "acct": "2201269",
    "addr": "13923 GREENLAND AVE NW, Lake Twp., OH, 44685"
    },
    {
    "acct": "2205083",
    "addr": "13875 GREENLAND AVE NW, Lake Twp., OH, 44685"
    },
    {
    "acct": "2200881",
    "addr": "13905 GREENLAND AVE NW, Lake Twp., OH, 44685"
    },
    {
    "acct": "2201270",
    "addr": "13941 GREENLAND AVE NW, Lake Twp., OH, 44685"
    }
    ],
    "picklist": [
    {
    "name": "action_text",
    "value": ""
    },
    {
    "name": "adv_weather_om",
    "value": "Unselected"
    },
    {
    "name": "breaker_text",
    "value": ""
    },
    {
    "name": "cause_om",
    "value": "Primary Mesh"
    },
    {
    "name": "control_zone_text",
    "value": "OPAL Stark
    "name": "office_text",
    "value": ""
    },
    {
    "name": "other_cause_om",
    "value": "Overload"
    },
    {
    "name": "primary_cause_om",
    "value": "Foreign Interference"
    },
    {
    "name": "remedy_om",
    "value": "Equipment Removed"
    },
    {
    "name": "scheduled_dev_om",
    "value": "Unselected"
    },
    {
    "name": "system_om",
    "value": "Transmission OH"
    },
    {
    "name": "trans_text",
    "value": ""
    },
    {
```

```
Lake Lake Sub 2414"
    },
    {
    "name": "def_equip_om",
    "value": "Customers Equipment"
    },
    {
    "name": "device_text",
    "value": "T11449"
    },
    {
    "name": "district_text",
    "value": ""
    },
    {
    "name": "division_text",
    "value": ""
    },
    {
    "name": "emerg_sw_order_text",
    "value": ""
    },
    {
    "name": "est_out_om",
    "value": ""
    },
    {
    "name": "failure_om",
    "value": "Unknown"
    },
    {
    "name": "feeder_text",
    "value": ""
    },
    {
    "name": "foreign_interf_om",
    "value": "Unselected"
    },
    {
    "name": "human_elem_om",
    "value": "Unselected"
    },
    {
    "name": "interrupt_dev_om",
    "value": "Substation Breaker"
    },
    {
{
"accountId": "6386568150",
"notificationType": "ZZ-UOUT",
"messageId": "2088",
"origin": "NMS",
"notificationTemplate": "",
"data": [
    {
    "key": "CUSTOMERADDRESS1",
    "value": "10375 Newbury Ave NW,Uniontown, OH 44685"
    },
    {
    "key": "CAUSE",
    "value": "TROUBLE_CALL"
    },
    {
    "key": "CREWETA",
    "value": "0"
    },
    {
    "key": "OUTAGETIME",
    "value": "1646921959"
    },
    {
    "key": "ESTRESTTIME",
    "value": "1646926759"
    },
    {
    "key": "COMPLETIONTIME",
    "value": "-1"
    },
    {
    "key": "CUSTOMERSOUT",
    "value": "1"
    }
]
}
```

```
"id": 9005,
```

NMS Notification Message (JSON)	SourceNotificationStream (JSON)

```
                "name": "type_om",
                "value": "Unselected"
                },
                {
                "name": "vegetation_om",
                "value": "Unselected"
                }
                ]
                }
        ]
        }
```

Sample OU CCB Notif MetaData Sync Message

Message Received from CCB API Call (C1-TemplateBasedNotifTypes)

```
{{
    "notificationTypes": {
        "notificationType": [
            {
                "notificationType": "BILL_DUE_TMPL",
                "notificationPushSub": "C1SU",
                "isCustomerContactRequired": false,
                "overrideDoNotDisturb": false,
                "hasOverrideDelivery": false,
                "descriptions": {
                    "description": [
                        {
                            "description": "Bill Due (Template Based)",
                            "language": "ARA"
                        },
                        {
                            "description": "Bill Due (Template Based)",
                            "language": "ENG"
                        },
                        {
                            "description": "Bill Due (Template Based)",
                            "language": "ESA"
                        },
                        {
                            "description": "Bill Due (Template Based)",
                            "language": "FRA"
                        },
                        {
                            "description": "Bill Due (Template Based)",
                            "language": "POL"
                        },
                        {
                            "description": "Bill Due (Template Based)",
                            "language": "ZHS"
                        }
                    ]
                },
                "deliveryTypes": {
                    "deliveryType": [
                        {
                            "deliveryType": "EMAI",
                            "isDefaultEnabled": false,
                            "templateNames": {
                                "templateName": [
                                    {
                                        "language": "ENG",
                                        "templateName": "Bill Due - Email"
                                    }
                                ]
                            }
                        },
                        {
                            "deliveryType": "SMS",
                            "isDefaultEnabled": false,
                            "templateNames": {
                                "templateName": [
                                    {
                                        "language": "ENG",
                                        "templateName": "Bill Due - SMS"
                                    }
                                ]
                            }
                        }
                    ]
                }
            },
        ]
    }
}
```