# Oracle® SDP Number Portability

Concepts and Procedures

Release 11i

August 2000

Part No. A86276-01

ORACLE®

Oracle SDP Number Portability Concepts and Procedures, Release 11i

Part No. A86276-01

# Contents

## Using Oracle Number Portability

## Administering Oracle Number Portability

# Send Us Your Comments

**Oracle SDP Number Portability Concepts and Procedures, Release 11i**

**Part No. A86276-01**

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us via the postal service.

Oracle Corporation
CRM Content Development Manager
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

---

---

---

If you have problems with the software, please contact your local Oracle Support Services.

# Preface

Welcome to the Oracle SDP Number Portability, Release 11i.

This Detailed Implementation Guide provides information and instructions about the implementaion of the Oracle SDP Number Portability application.

This preface explains implementation considerations and processes is organized and introduces other sources of information that can help you.

## Intended Audience

This guide is aimed at the following users:

- Technical Service Representatives (TSRs)
- Customer Service Representatives (CSRs)
- System Administrators (SAs), Database Administrators (DBAs), and others with similar responsibility.

This guide assumes you have the following prerequisites:

- Understanding of the company business processes.
- Knowledge of products and services as defined by your marketing policies
- Basic understanding of Oracle and Developer / 2000
- Understanding of the interface protocol to each of the fulfillment elements (telnet, script)
- Background in SQL, PL / SQL, SQL* Plus programming

## Structure

This manual contains the following chapters:

"Understanding Number Portability" provides overviews of the application and its components, explanations of key concepts, features, and functions, as well as the application's relationships to other Oracle or third-party applications.

"Using Number Portability" provides process-oriented, task-based procedures for using the application to perform essential business tasks.

"Administering Number Portability" provides task-based procedures for required for ongoing system maintenance and includes information on administration tools and utilities.

## Related Documents

For more information, see the following manuals:

- *Oracle SDP Number Portability Technical Reference Manual*
- *Oracle SDP Provisioning Concepts and Procedures*
- *Oracle SDP Provisioning Technical Reference Manual*

## Conventions

The following conventions are also used in this manual:

| Convention | Meaning |
|---|---|
| .<br>.<br>. | Vertical ellipsis points in an example mean that information not directly related to the example has been omitted. |
| . . . | Horizontal ellipsis points in statements or commands mean that parts of the statement or command not directly related to the example have been omitted |
| **boldface text** | Boldface type in text indicates a term defined in the text, the glossary, or in both locations. |
| < > | Angle brackets enclose user-supplied names. |
| [ ] | Brackets enclose optional clauses from which you can choose one or none. |

# Understanding Oracle Number Portability

This topic group provides overviews of the application and its components, explanations of key concepts, features, and functions, as well as the application's relationships to other Oracle or third-party applications.

## Number Portability Defined

Number portability enables users of telecommunication services on one switching system to move or "port" numbers to a different switching system.

Oracle Number Portability provides the capability for consumers to keep their telephone numbers when:

- Switching between telecommunication service providers
- Moving from one physical location to another
- Changing one telecommunication service for another telecommunication service

### Routing Numbers

A routing number makes the porting of numbers between switching systems possible. The routing number is mapped to the new telecommunication switching system on the one end and linked to the customer's current dialing number on the other end. Each service provider allocates certain numbers as routing numbers for number portability.

Every service provider using Oracle Number Portability has a routing number repository for this purpose. The table following lists the locations where routing numbers are recorded and the purpose of each location.

***Routing Number Repositories***

| Use | Location |
|---|---|
| Execution | The service provider's local database |
| Control | Central databases maintained by national or regional number centers. These are known as NRCs in the United States. They are also known as NPAC (United States), S-NPAC (Sweden), and CRDC (Belgium).<br>This system is generically referred to as NRC in Oracle's Number Portability product. |

# Types of Number Portability

The table following displays the three commonly accepted types of number portability.

***The Different Types of Number Portability***

| Type | Description |
|---|---|
| Service provider portability | The ability to retain the same telephone numbers as the user changes from one service provider to another.<br>For example, Dr. Jon Smith moves local telephone service for his office from Euro Telecom to the new emerging carrier SuperTel. The physical location of Dr. Smith's office has not changed. |
| Location portability | The ability to retain existing telecommunication numbers without impairment of quality, reliability, or convenience when moving from one physical location to another.<br>For example, Dr. Jon Smith moves his office from Brussels to Gothemburg and retains his existing telecommunications numbers with SuperTel. |
| Service portability | The ability to retain existing telecommunication numbers without impairment of quality, reliability, or convenience when switching from one telecommunications service to another telecommunication service provided by the same service provider.<br>For example, Dr. Jon Smith moves a subset of the telecommunications numbers for his office from wireline service to wireless service with SuperTel. When signing up for wireless service, he retains the existing telecommunications numbers with the same set of features that he had with wireline service. |

## Understanding Orders

There are a number of order activities that you carry out in Oracle SDP Number Portability. The following table lists the types of order activity that you can perform in the application and provides a short description of each.

***Examples of Order Activity in Oracle Number Portability***

| Order Type | Description |
|---|---|
| Port In | A port in request is usually initiated at the recipient service provider side by a new customer requesting that you provide a port in capability. |
| Port Out | A port out request can occur in one of two ways:<br>■ It is triggered internally at the donor service provider side based on the same customer contact as at the recipient side.<br>■ It is possible for a customer to contact the donor service provider directly in order to initiate a port out request to another service provider. |

**Note:** Standard activities are provided for the user to build business processes for each order type. These order types do not come seeded with the product.

## Ways to Submit a Porting Request

There are two ways to submit a porting request to Oracle SDP Number Portability. The table following describes each.

***Ways to Submit a Porting Request***

| Submission Type | Description |
|---|---|
| You use the application graphical interface. | Information is entered through the Port In Request window, accessed via the Test Center. |
| You use the APIs provided with the application. | This include the following APIs:<br>■ Process Order API<br>■ Submit Order API<br>■ Synchronization Order API |

# Porting Phases in Oracle Number Portability

There are four porting phases in Oracle Number Portability. The following table provides a description of each.

***Porting Phases in Oracle Number Portability***

| Phase | Description |
|---|---|
| Inquiry | A customer calls the new service provider and inquires as to the feasibility of porting his/her number.<br>For example, Dr. John Smith may call three new service providers to ask what the rates would be if he ported his number.<br>Note that it is possible for multiple porting inquiry requests to be placed for the same telephone number at the same time. |
| Ordering | A customer calls and requests service from a new service provider and asks to keep his/her telephone number from the old service provider.<br>Only a single porting order request can be placed for a telephone number at any given time. For example, Dr. John Smith can only select one service provider to port his number. |
| Active | A customer's telephone number is currently active in the network with the new service provider.<br>Note that the Active phase should not be confused with the Active flag. The Active flag is defined for each porting status to indicate whether the porting status is a valid value that can be used by the application. |
| Old | A porting request that has been canceled or disconnected, etc.<br>It is suggested that, after a specified time period, porting requests with a status in the Old phase should be archived or purged. |

Typically, the phase of the service order in the number porting process most often determines the tasks that you perform with a service order (but not always).

For example:

- You perform the tasks associated with entering orders in the Ordering phase of number porting.
- You perform monitoring and number range maintenance and their component tasks in any phase of a number porting process.

# Porting Statuses in Oracle Number Portability

Porting statuses are used to track the progress of a porting request.

A service order may have more than one status throughout its life cycle. You can define a new status through Oracle SDP Number Portability for either of the following reasons:

- To better reflect the terminology and business processes of your organization
- To understand the progression of a porting request

Once a status is defined it can be referenced in Workflow and process logic. These are the places where manipulation of the statuses occur.

### Key Components of a Porting Status Definition

In Oracle SDP Number Portability, each porting status is user-definable. It must also be associated with one of the pre-defined phases of a porting request. User-defined porting statuses provides flexibility to users (for example, naming conventions are different for each country). Associating a porting status with a pre-defined porting phase allows applications to categorize porting statuses, adding to the manageability of porting requests.

The porting phases are:

- Inquiry
- Ordering
- Active
- Old

In effect, porting statuses enable you to more closely map your business processes to the progress of a service order than is possible with only the four phases.

## Workflow in Oracle Number Portability

Workflows in Oracle SDP Number Portability are sub-processes which execute during the order fulfillment process.

- Workflows are used to automate the business processes necessary to fulfill the order.
- Workflows are generally referred to as work items in the application.
- Workflows are created using the Oracle Workflow Builder.

### Oracle Workflow Builder

You use the Oracle Workflow Builder to customize your business needs. Through workflow, you can route any type of information in an asynchronous manner, according to your business rules.

Within the Oracle Workflow Builder, you create, view, or modify a business process with simple drag and drop operations. In addition, you can create and modify all workflow objects, including activities, item types, processes and notifications.

At any time, you can perform the following operations on workflow objects:

- Add
- Remove
- Modify
- Set up new prerequisite relationships among the various types

You can easily work with a summary-level model of your workflow, expanding activities within the workflow as needed to greater levels of detail.

### Reference

See the Oracle Workflow documentation set for information relating to the use of workflow.

## Workflow Activity Functions

When defining a workflow, it is important to understand the interaction between Oracle Workflow and Oracle SDP Number Portability.

In simple terms, you build your workflow using any or all of the following types of function activities:

- Standard function activities provided by Oracle Workflow
- Standard function activities provided by Service Delivery Platform
- Standard function activities provided by Oracle SDP Number Portability
- Your own user-defined function activities

### Standard Function Activities Provided by Oracle Workflow

These types of function activities are actions that simply progress a workflow to the next activity, or change standard workflow information. For example, this could be merely the start or the end of a workflow.

- These functions are defined during workflow installation, in the Standards section.
- These procedures types are held in the WF_STANDARD package in any Oracle database.
- Do not adjust these procedures.

### Standard Function Activities Provided by the Service Delivery Platform

Function activities in workflow are split into those activities specific only to the Service Delivery Platform, and those activities specific only to number porting.

- Function activities that are specific to the Service Delivery Platform are generally related to messaging, timers, synchronization, preparing notifications, work items, and fulfillment actions.
- Function activities that are specific to number porting are related to the number portability process.

This allows for a better visual understanding of the difference between the activities.

Certain actions that take place in a process require a check with, or an update to, the Service Delivery Platform configuration.

Several examples:

- If an activity in a workflow is to change the porting status of an order, a number of checks need to be made against the data held in Service Delivery Platform. In particular, the application must verify that the new porting status is a valid one.
- During Send Message a check must be made with the application's repository to ensure that the message is a valid message and that the parameters are correct.

The actual PL/SQL procedures for functions that are specific to Service Delivery Platform and Oracle SDP Number Portability are held in the XNP_WF_STANDARD package.

The table following lists the Service Delivery Platform standard function activities supplied with the application.

***Service Delivery Platform Standard Function Activities***

| Name | Description |
|---|---|
| Check Order Result | Determines the value of the of the ORDER_RESULT work item parameter, and completes the activity based on this value. |
| Complete Work Items and Update Status | Sets the work item status, and notifies the order tracking system (the Service Delivery platform), about the completion of the work item. |
| Execute Fulfillment Action | Executes the fulfillment action for the given feature type. It associates the fulfillment element to a work item, and submits it for provisioning. It then registers for an event to wake it up after the fulfillment action is complete. |
| Prepare Customized Notification | Defines a customized notification to be displayed at runtime. |
| Prepare Notification Message | Prepares the notification to be sent to the target. |
| Publish Event | Publishes a single business event. In the case of internal events, any recipients of this event must already have subscribed to the event. |
| Send Message | Sends a message to a single recipient.<br>The Send procedure first checks whether or not an adapter is available for the recipient of the message.<br>■ If an adapter is available, it proceeds with the send action.<br>■ If an adapter is unavailable, a callback is registered to receive a notification once the adapter becomes available. |
| Set Order Result | Sets the value of the ORDER_RESULT work item parameter to the lookup value set by the workflow builder. |
| Subscribe to Acknowledgments | Identifies and registers for all expected responses messages. (In many cases, sending a request message can result in the generation of multiple response messages.) |
| Subscribe to Business Event | Registers a callback for the given event from the remote or local system. |

## Standard Function Activities Provided by Oracle SDP Number Portability

The table following lists the Number Portability standard function activities supplied with the application.

***Number Portability Standard Function Activities***

| Name | Description |
|---|---|
| Check if Donor Can Port Out | Checks whether or not the donor service provider of this porting transaction has provisioned the number range, or has assigned the number range.<br>■ If either condition is true, then the activity completes with Y. |
| Check if Donor is Initial Donor Also | Checks whether or not this donor is also the initial donor. |
| Create Porting Order | Creates a Porting Record for each telephone number in the range. The Porting ID for the first record is the same as that set in the work item parameter. |
| Create SMS Porting Records | Creates a Porting Record for each telephone number in the range. The Porting ID for the first record is the same as that set in the work item parameter. |
| Deprovision FEs | Sets the fulfillment elements to be deprovisioned for this feature type and number range. For each fulfillment element, the Service Delivery Platforms's provisioning procedure (Execute FA) is invoked.<br>At the end of this activity, control passes to the provisioning subsystem which executes the fulfillment procedure. An FA_DONE message is subscribed for each fulfillment action being executed which gives the execution result of the fulfillment procedure. The callback procedure associated with the FA_DONE handles the responses received.<br>The immediate next activity following this activity must be the SDP Standard **Wait For Flow**. This is to ensure proper hand-off from the provisioning system back to the Number Portability system.<br>Only fulfillment elements that were earlier provisioned by this service provider can be modified. Otherwise, the fulfillment elements are ignored. |
| Determine Current Service Provider Role | Determines whether or not a given service provider is the donor, original donor or recipient for the current porting transaction and completes the activity with the appropriate result code. |
| Determine If Subsequent Porting Request | Checks whether or not this is a subsequent porting request.<br>■ Returns either Y or N. |
| Does Porting Record Exist for Donor | Check whether or not there exists a porting record with the given status in this telephone number range that belongs to the given donor's service provider ID.<br>■ Returns either Y or N. |

***Number Portability Standard Function Activities***

| Name | Description |
|---|---|
| Does Porting Record Exist for Recipient | Check whether or not there exists a porting record with the given status in this telephone number range that belongs to the given recipient's service provider ID.<br><br>Returns either Y or N. |
| Get Flag Value | Retrieves the Locked flag value for the given PORTING_ID work item parameter.<br><br>■ The activity completes with a flag value of Y or N. |
| Get Porting Status | Retrieves the status of the porting record for the given Porting ID |
| Modify FEs | Retrieves the fulfillment elements to be modified for this feature type and number range. For each fulfillment element, the Service Delivery Platform's provisioning procedure (Execute FA) is invoked.<br><br>At the end of this activity, control passes to the provisioning subsystem which executes the fulfillment procedure. An FA_DONE message is subscribed for each fulfillment action being executed which gives the execution result of the fulfillment procedure. The callback procedure associated with the FA_DONE handles the responses received.<br><br>The immediate next activity following the activity must be SDP Standard **Wait For Flow**. This is to ensure proper hand-off from the provisioning system back to the Number Portability system.<br><br>Only fulfillment elements that were earlier provisioned by this service provider can be modified. Otherwise, the fulfillment elements are ignored. |
| Provision FEs | Sets the fulfillment elements to be provisioned for this feature type and number range. For each fulfillment element, the Service Delivery Platform's provisioning procedure (Execute FA) is invoked.<br><br>At the end of this activity the control passes to the provisioning subsystem which executes the fulfillment procedure.<br><br>An FA_DONE message is subscribed for each fulfillment action being executed which gives the execution result of the fulfillment procedure. The callback procedure associated with the FA_DONE handles the responses received.<br><br>The immediate next activity following this activity must be the SDP Standard **Wait For Flow**. This is to ensure proper hand-off from the provisioning system back to the Number Portability system. |
| Reject Message | Rejects a message. |
| Remove from SMS Provisioning Map | Deletes the fulfillment element mapping for this telephone number range from the application database. |
| Retry Message | Retries the message. |

***Number Portability Standard Function Activities***

| Name | Description |
|---|---|
| Set Flag Value | Sets the flag to the given value for the entities in XNP_SV_SOA for the given PORTING_ID work item parameter and FLAG_NAME for the current service provider.<br>■ Sets a values of Y or N. |
| Update Charging Information | Updates the Subscription Version in the Service Order Administrator for each telephone number with the given Porting ID with the invoice information. |
| Update Comments and Notes Information | Updates the comments and notes for the current Porting ID and the current service provider. |
| Update Current SV Status | Updates the status type code in the XNP_SV_SOA with the new status type code. All records with the Porting ID and belonging to the current service provider are updated to the new status.<br>If the new status belongs to the ACTIVE phase, and if there exists records for this number range already in ACTIVE phase, then these records are first reset to the OLD phase. The actual update of the records with the given porting ID is performed after this step. |
| Update Customer Information | Updates the customer information for the current Porting ID and the current service provider. |
| Update Network Information in SOA | Updates the network information in the XNP_SV_SOA for the current Porting ID and the current service provider. |
| Update Porting Status | Updates the status type code in the XNP_SV_SOA with the new status type code. All records with the porting ID and belonging to the current service provider are updated to the new status.<br>If the new status belongs to the ACTIVE phase, and if there exists records for this number range already in ACTIVE phase, then these records are first reset to the OLD phase. The actual update of the records with the given porting ID is performed after this step. |
| Update SMS Provisioning Map | Updates the Provisioning Status of the fulfillment element for the given feature type and telephone number range. |
| Update with New Date | Updates the date for the porting record with the given Porting ID.<br>The format used must be in the following format:<br>■ YYYY/MM/DD HH24:MI:SS |
| Verify Porting Status | Checks if the STATUS_TYPE_CODE from XNP_SV_SOA for the given PORTING_ID is same as the given status type code (in STATUS_TO_COMPARE_WITH).<br>■ Returns T if the two statuses match.<br>■ Returns F if the two statuses do not match. |

### User-defined Function Activities

Some actions that take place in a process are user specific actions. These actions may, or may not, have standard workflow or Oracle SDP Number Portability implications.

For example, it is possible that a number in a log table must be incremented every time a status change has occurred. As this activity is not an order specific action, this is merely an example of a procedure that retrieves the number from the table and adds one to it.

In contrast, a user-defined procedure contains some workflow or Oracle SDP Number Portability API calls in it.

For example, returning to the previous log table example, it is possible that there is a requirement to log the From Status and the To Status when an order status change takes place.

In order to perform this action, a user-defined procedure must first retrieve the order parameters or the work item parameters using an Oracle SDP Number Portability API call, and then insert them into the log table.

Another example of a user-defined procedure that is referenced by a workflow function is the process of retrieving the Existing Service Provider Name parameter from the Oracle SDP Number Portability order and then using the retrieved value to determine the service provider identifier. The procedure code can then update the work item parameter SP_ID in the Oracle SDP Number Portability order.

## Work Items in Oracle Number Portability

Oracle Number Portability makes use of workflows to automate business processes in the same manner as does Oracle Provisioning. Work items perform the business and network functions necessary to fulfill a service order request.

However, in the application each work item typically maps to a single workflow, due to the complexity of the business process. That is why workflows are referred to as work items in Oracle SDP Number Portability.

- The workflows delivered with the application were created using the Oracle Workflow Builder. You need to use this application to customize the workflow to meet your business needs.
- Work items comprise function activities. You may need to modify some or all of these while customizing the work items for your business.

The Oracle Workflow installation comes with a standard set of pre-defined notifications and function activities to use in building business processes. The application provides an additional set of pre-defined stub processes, notifications and function activities to support messaging and the porting process. The stub processes must be customized using Oracle Workflow Builder.

## Predefined Work Items in Oracle Number Portability

Work items are organized into groups known as item types. The table following lists the standard item types delivered with Oracle Number Portability. Do **not** modify any of the standard Oracle SDP Number Portability item types, except NP Processes.

***Standard Work Item Types in Oracle Number Portability***

| Item Type | Internal Name | Description | Customizable |
|---|---|---|---|
| Standard | WFSTD | Contains standard function activities that are commonly used to create business processes. | No |
| SDP Standard | XDPWFSTD | Contains function activities that are commonly used across both the Oracle Provisioning and Oracle Number Portability applications.<br>It includes, for example, the following functions:<br>Send Message<br>Subscribe to Acknowledgments | No |
| SDP Lookup Code | XDPCODES | Contains lookup types used for easy selection in workflow activities.<br>This could be, for example, a list of fulfillment actions, or a list of messages. | No |
| NP Standard | XNPWFSTD | Contains function activities that are specific to Oracle SDP Number Portability. Use these activities in the implementation of your business processes.<br>Examples of standard functions are the following:<br>■ Creating a porting order<br>■ Determining if porting is possible | No |

***Standard Work Item Types in Oracle Number Portability***

| Item Type | Internal Name | Description | Customizable |
|---|---|---|---|
| NP Lookup Code | XNPTYPES | Contains lookup types used for easy selection in workflow activities.<br>This could be, for example, a list of fulfillment actions, or a list of messages. | No |
| NP Processes | XNPWFACT | Contains country-specific number portability function activities that you customize for different implementations of the application within the same country.<br>It includes, for example, the following processes:<br>■ Process porting inquiry<br>■ Process porting order<br>■ Provision network element | Yes |

### Guidelines

1. In general, you should use the SDP Standard and NP Standard function activities whenever possible, and only customize the Number Portability activities as absolutely necessary.
2. SDP Lookups and NP Lookups must be updated as you start to configure Service Delivery Platform. Run the Lookups Loader API to update these item types.

### Reference

Consult the *Developing Oracle SDP Number Portability Applications Reference Guide*, for details on these item types.

## Standard Work Item Parameters

Business processes in Oracle SDP Number Portability are driven by messages or events. Messages can be exchanged between two different service providers or between different components of the same service provider.

- To maintain the state or context of the business process, each work item or workflow must include the set of standard work item parameters in addition to any business specific parameters. The developer must explicitly assign standard work item parameters to the work item.

- Work item parameters act as a parameter pool for all the defined work items in Oracle SDP Number Portability. You define new work item parameters using the existing parameters in the pool. In addition, you can add parameters to the pool to define new work item parameters, or to extend existing work item parameters assigned to a work item.

## Mandatory Work Item Parameters

The standard work item parameters listed in the table following are **mandatory** for any Oracle SDP Number Portability work item. All standard parameters are required for any message header that is generated, except the **Messaged** parameter. The Message_ID is used to identify the most recent incoming or outgoing message for a work item instance. This is essential when managing messages (for example, as in re-sending a message).

> **Note:** Oracle SDP Number Portability Work Item Parameters are shown in the application as the Display Name. Sometimes the display name differs from the parameter name. For example, **SP_Name** displays in the application as **Owning Service Provider**.

***Mandatory Work Item Parameters***

| Parameter Name | Description |
|---|---|
| SP_NAME | Code of the service provider for which the work item is executing.<br>Display Name: Owning Service Provider |
| OPP_REFERENCE_ID | The reference identifier used on the peer system. All responses to peer requests should populate this field from the REFERENCE_ID field of the request message.<br>Display Name: Opp Reference ID |
| SENDER_NAME | Code of the sending service provider. This is included in all messages sent from the local system to the peer system.<br>Display Name: Sender Name |
| RECIPIENT_NAME | Code of the receiving service provider. This is included in all messages sent from the local system to the peer system.<br>Display Name: Recipient Name |

***Mandatory Work Item Parameters***

| Parameter Name | Description |
|---|---|
| MESSSAGE_ID | Identifier of the newly constructed message. The message text is retrievable using the identifier. |
| | Display Name: Message ID |
| STARTING_NUMBER | Starting telephone number in a Service Delivery Platform order. |
| | Display Name: Starting Number |
| ENDING_NUMBER | Ending telephone number in an Service Deliver Platform order. |
| | Display Name: Ending Number |

# Default Work Items in Oracle SDP Number Portability

**Note:** These work items are optional. You chose whether or not to load the default work items during application installation.

The table following provides a summary of the seeded work items provided with Oracle SDP Number Portability. These work items need to be customized to meet the needs of your installation of the application.

***Work Items Seeded by Oracle Number Portability***

| Work Item | Internal Name | Description |
|---|---|---|
| Cancel Disconnect Porting Request from Order Entry | CANCEL_DISC_PORT_REQ_FROM_OMS | Customer contacts current operator to cancel a disconnect subscription request. |
| Cancel Modify Porting Request from Order Entry | CANCEL_MODIFY_PORT_FROM_OMS. | Customer contacts new operator to cancel a modification previously submitted on a porting request. |
| Cancel Porting Request from Order Entry | CANCEL_PORT_REQ_FROM_OMS | Customer contacts new operator to cancel a porting request. |
| Charge New Operator for Porting Request | REC_RECEIVE_CHARGING_NOTIF. | Donor Operator charges recipient operator for the port-out transaction. |

***Work Items Seeded by Oracle Number Portability***

| Work Item | Internal Name | Description |
|---|---|---|
| Create or Modify Ported Number(s) | PROVISION_PORTED_NUMBER. | Create or Update network elements with new porting data received from Number Registration Center. |
| Delete Ported Number(s) | DELETE_PORTED_NUMBERS. | Remove porting data from network elements provisioned earlier. This may be in response to a broadcast received from Number Registration Center to carry out this activity. |
| Disconnect Porting Request from Order Entry | DISC_PORT_REQ_FROM_OMS | Customer contacts current operator to disconnect subscription. |
| Hold Porting Request from Order Entry | HOLD_PORT_REQ_FROM_OMS. | Customer contacts new operator to place a porting request on hold. |
| Hold Porting Request from other Operator | HOLD_PORT_REQ_FROM_OPERATOR. | Hold Porting Request received from other operator. |
| Inquire Donor Operator for Porting Out | PORTING_INQUIRY_FROM_OPERATOR. | Donor operator determines whether porting inquiry request should be approved or rejected. |
| Inquire Recipient Operator for Porting In | PORTING_INQUIRY_FROM_OMS | New Operator receives porting inquiry from customer care. |
| Load, Disaster Recovery & Backup of Local Database | LOAD_DISASTER_RECOVERY_BACKUP | Synchronize local database with Number Registration Center. |
| Modify Porting Request from Operator | MODIFY_PORT_REQ_FROM_ OPERATOR. | Modifies the Porting Request on receiving such a request from the other operator. |
| Modify Porting Request from Order Entry | MODIFY_PORT_REQ_FROM_OMS. | Customer contacts new operator to modify a porting request. |
| Number Range Split Request from Order Entry | NUMBER_RANGE_SPLIT | Number Range Split declared by a regulatory board. |
| Porting Notification Concurrence | RECEIVE_CONCURRENCE | Donor Operator and Recipient Operator receive concurrence notification from Number Registration Center for a porting transaction. |
| Porting Order initiated by Recipient Operator | PORTING_ORDER_FROM_OMS | Recipient Operator receives porting order from Customer Care. |

***Work Items Seeded by Oracle Number Portability***

| Work Item | Internal Name | Description |
|---|---|---|
| Query Porting Data | QUERY_REFERENCE_DATA. | Query Number Registration Center for porting data. |
| Reject Porting Request | PORTING_NOTIFICATION_REJECTION. | Donor Operator and Recipient Operator receive rejection notification from Number Registration Center for a porting transaction. |
| Remind Operator for Porting Response | PORTING_NOTIFICATION_REMINDER. | Number Registration Center reminds operator that a response is required to proceed with the porting process. |
| Respond to Porting Order received by Donor Operator | PORTING_ORDER_FROM_OPERATOR. | Donor Operator responds to a Porting Request. |
| Transfer Number Range Holder from Order Entry | TRANSFER_NUMBER_RANGE_HOLDER | Transfer Number Range Holder declared by a regulatory board. |

# Adapters in Oracle SDP Number Portability

You configure an adapter to interact with the external system and to route messages to and from the external system appropriately. You configure an adapter by specifying its attributes.

The application comes with the file adapter already defined. This adapter enables messages to be passed to local files, or sent to a remote location using the provided FTP client.

## Outbound Messages

For outbound messages, the adapter receives the message from the Oracle Advanced Queue, and generates the channel information accordingly.

According to the need, the adapter sends the message to one of the following channels:

- A local file
- A remote system (using File Transfer Protocol)

### Inbound Messages

For inbound messages, the adapter acts like a server. It retrieves the message by monitoring a designated file directory. It then pushes the messages in the files it retrieves to the queue. MSG_SERVER is the consumer for these inbound messages.

### Channels

A channel defines the path a messages takes during message transfer from, or to, a clear destination. You can only have one channel open per message at any given time. If you want to operate a new channel, then you must close the old channel. For example, if you open a file and send messages to it, then you must close that file before you can open another file channel.

### File Adapter Attributes

The table following lists the attributes you must define for a file adapter.

***File Adapter Attributes***

| Display Name | Internal Name | Description |
|---|---|---|
| Home Directory | HOMEDIR | The directory on the local machine where the file is to be created. |
| In Directory | IN_ARCHIVE_DIR | The directory to which the file is to be archived after the adapter reads it. |
| IP Address | IP_ADDRESS | IP address of the remote machine to which the file is to be sent. |
| Out Directory | OUT_ARCHIVE_DIR | The directory in which the file is to be archived after the file transfer has completed. |
| Password | PASSWORD | Password for authentication. |
| PORT | PORT | The port on which the remote FTP server is running. The standard port is 21. |
| Remote Directory | REMOTEDIR | The directory on the remote machine where the file is to be placed. |
| Scan Directory | SCANDIR | The directory from which files are read and sent to the application for processing. |
| User ID | USER_ID | The user ID for the authentication. |

# Fulfillment Element Types in Oracle Number Portability

Every fulfillment element belongs to a fulfillment element type group. The application supports the creation of new element types, if necessary.

Typically, you need fulfillment element types that correspond to the following categories:

- SCP (service control point)
- SSP (signal switching point)
- STP (signal transfer point) types
- SP (service providers) gateways
- NRC (number registration center) gateways

## Predefined Fulfillment Element Types

The Oracle Number Portability application comes with a number of predefined fulfillment element types. The table following lists these predefined element types.

***Predefined Element Types in Oracle Number Portability***

| Element Type | Description |
|---|---|
| BILLING_GATEWAY | Refers to billing system gateway |
| DIRECTORY_GATEWAY | Refers to Directory assistance gateway |
| NRC_GATEWAY | Refers to the Number Registration Center gateway |
| SCP | Refers to the LNP database for all service control points |
| SP_GATEWAY | Refers to messaging gateways for remote systems of other Service providers |
| SSP | Refers to signal switching point |
| STP | Refers to signal transfer point |

If the fulfillment element is a network element of type SCP, SSP, or STP, then you must define the number ranges served by this network element for a given feature type.

**Note:** If the fulfillment element is a network element of type SCP, SSP, or STP, then you must define the number ranges served by this network element for a given feature type.

# Fulfillment Elements in Oracle Number Portability

A fulfillment element is either one of the following:

- A network element
- An external system

For example, Service Switching Points and Signaling Transfer Points are types of network elements typically provisioned for number portability.

## Fulfillment Elements as Network Elements

You configure each network element as a fulfillment element. This process includes specifying the following items for the fulfillment element.

- Its network ID
- Its type
- Its attributes
- Its software versions
- The supported communication protocols
- The adapter type

## Fulfillment Elements as External Systems

Every service provider (external system) with which the application interacts must have a corresponding fulfillment element created for it. This fulfillment element represents the configuration setup for the gateway to that external system.

- You **must** define the appropriate fulfillment element in order for the application to interact with an external system.
- You **must** configure adapters that can interact with that external system. Adapters route messages sent to and from the external system appropriately.

# Understanding Fulfillment Actions

Fulfillment actions apply fulfillment procedures to fulfillment elements.

Each fulfillment element has a software version and an adapter assigned to it, and is of a certain fulfillment element type. From this information, the fulfillment action determines which fulfillment procedure to perform on this fulfillment element.

- One fulfillment action is written for every action that is performed on a fulfillment element.
- A fulfillment action acts upon only a single fulfillment element. (This is true at runtime only.)
- A fulfillment action can act on multiple fulfillment element types, however, by specifying for each element type the adapter, the software version, and the procedure to be applied.
- If fulfillment actions are defined for use in Oracle Number Portability, then they are assigned internal names such as Provision Number Portability, Deprovision Number Portability, or Modify Number Portability. Display names correspond to these internal names. A user may define fulfillment actions, as well.

## Predefined Fulfillment Actions

Oracle Number Portability comes with a number of fulfillment actions predefined. In most cases, it is not necessary to define new actions. These predefined fulfillment actions are listed in the following table.

***Fulfillment Actions Predefined in Oracle Number Portability***

| Deprovision | Provision | Modify |
|---|---|---|
| DEPROVISION_CLASS | PROVISION_CLASS | MODIFY_CLASS |
| DEPROVISION_CNAM | PROVISION_CNAM | MODIFY_CNAM |
| DEPROVISION_ISVM | PROVISION_ISVM | MODIFY_ISVM |
| DEPROVISION_LIDB | PROVISION_LIDB | MODIFY_LIDB |
| DEPROVISION_NP | PROVISION_NP | MODIFY_NP |
| DEPROVISION_RN | PROVISION_RN | MODIFY_RN |
| DEPROVISION_WSMSC | PROVISION_WSMSC | MODIFY_WSMSC |

## Understanding Fulfillment Procedures

A fulfillment procedure is composed of specific commands that are sent to a fulfillment element when a fulfillment action is invoked. Examples of such commands might be to open a Telnet session or to provide element-specific routing commands to the element.

A fulfillment procedure comprises the following elements:

- A fulfillment element type
- An adapter
- The software version to be used on this fulfillment element type
- The fulfillment action to be performed

## Messaging in Oracle SDP Number Portability

Oracle SDP Number Portability uses messages to communicate with external systems and initiate new orders in Service Delivery Platform. For example:

- Oracle SDP Number Portability sends an outgoing message requesting a number port with the Central System.
- The Oracle SDP Number Portability workflow then waits for an inbound message that confirms the request to port.

This use of messaging integrates the use of workflows with external conditional elements.

Messages can trigger a process, or set of processes in an application. Applications send and receive messages asynchronously using the Event Manager.

## Messages Defined

Messages are used to communicate between applications and systems. Service Delivery Platform messages are defined in the industry-standard XML format. When a message is compiled, the following functions are created:

- Send()
- Publish()
- Validate()
- Process()

- Default_Process()
- Create_Msg()

These functions are used to communicate with the Number Registration Center and other service providers as well as with any other external system or internal Operational Support Systems.

Also, messages can trigger a set of processes internally within the system.

### Message Definition

You define messages in one of two ways:

- You can select from a set of predefined and available messages that are preseeded in Oracle Number Portability.
- You can build and define any required messages through the iMessage Studio.

### Message Elements

A message comprises one or more message elements. A message element consists of a set of attributes:

- Type
- Name
- Length
- Datatype

For example, area may be a message element composed of length and width. The message element name is used as a tag within the XML message. Each message element has a data source which must be defined.

### Messages versus Events

The table following lists the differences between messages and events.

***Messages versus Events***

| Type | Description |
|---|---|
| Message | A request to, or a reply from, a single pre-defined destination |
| Event | Broadcast notification to zero or more destinations that need not be specified until runtime |

# How the Event Manager Handles Messages

The Event Manager handles all messages entering the application system. Incoming messages are one of the following types:

- Request messages
- Responses to request messages
- Event notifications from remote systems

Remote applications send request messages and register for response messages with the Event Manager. The remote applications use an Oracle SDP Number Portability API to register for messages.

When a message arrives, the Event Manager delivers the message to all registered applications after executing the validation and processing logic defined for the message. If no application has registered for a message, the application's default processing logic for that message is executed after message validation.

## Message Processing by the Event Manager

The Event Manger performs the following actions upon arrival of a new message in the system.

1. It first retrieves the message from the Inbound Message queue or the Internal Events queue.
2. It executes the validation logic for the message.
3. It then searches the callback registration table to determine all workflow instances registered for the received message code and reference ID combination.
   - If a match is found, the Event Manager executes the incoming processing logic for the message.
   - If no matches are found, the Event Manager executes the default processing logic for the message. It then retrieves the next message from the Inbound Message queue and starts the process over.
4. Finally, it attempts to deliver the message to the registered workflow instances. If the message is not successfully delivered, the message status is marked as failed for the registrant. This status can be seen in the Callback Registration data.

## Configuring iMessage Subscription

Within Oracle SDP Number Portability, you can set applications to be default subscribers to messages. This is configured in the iMessage Subscribers utility. You can also associate default subscribers with events. For example, when a message occurs, the Event Manager ensures that an outbound message is automatically sent to the subscriber identified by a fulfillment element.

### Automatic Responses

If desired, you can associate one or more responses with an event. A response is an acknowledgment to a message.

For example, valid message responses for the message "Is this an existing customer?" are:

- "Yes, this is an existing customer."
- "No, this is not an existing customer."

These responses are messages in themselves and must be configured in the application before they can be linked as responses to a message.

### Manually Driven Messaging

There are also a variety of APIs available in Service Delivery Platform to facilitate custom subscription, de-subscription, enqueueing and dequeueing. Consult the *Oracle Developing XML-based Message Based Applications* reference guide for a detailed definition of the available API calls.

# The iMessage Studio

Use the iMessage Studio as a tool for developing message-based Service Delivery Platform applications. It provides the following functionality:

- The ability to develop a message-based application, while generating the code to construct, publish, validate, and process application messages.
- The ability to compile and test these messages.
- The ability to share messages between applications, allowing for their re-use in various applications.
- The means to prevent redefining the same message in various applications across the enterprise.
- The ability to generate procedures through its APIs and its run-time messages.

- The means to customize pre-processing of outbound messages and the post-processing of incoming messages from the Event Manager to external subscribers.
- The ability to define a message set for a particular event during processing of an order within the application.
- The ability to construct timers (delayed messages) for use within the application.
- The ability to define messages in the XML industry-standard format.
- The ability to create a series of message processes to communicate with external systems, Service Providers, and internal Operational Support Systems.

## Messages, Events and Timers

You use the iMessage Studio to create the following message-based items:

- Messages
- Events
- Timers

The following table lists the three types and describes each one.

***Messages, Events and Timers***

| Type | Description |
|---|---|
| Messages | A sequence of text characters that are used for communication between application systems. Messages fall into two categories:<br>■ Messages for internal applications<br>Internal applications can register a PL/SQL callback procedure via the Event Publisher, or through an API.<br>■ Messages for external applications<br>External applications do **not** register callback procedures, but have adapters running to relay the published event to the remote system. External applications can register for an event using the default subscribers screen.<br>Oracle SDP Number Portability explicitly supports only the XML format. |
| Events | Messages that are sent to external systems and that are received from external systems. Events are published to both external and internal applications. |
| Timers | Messages that have a time delay and a duration interval associated with them. |

# Timers Defined

Timers in Service Delivery Platform are used to handle events or processes that must occur at specified time intervals within the application.

In general, timers are used in either one of two ways:

- To perform a task once, after a delay
- To perform a task repeatedly, after a delay

> **Note:** Service Delivery Platform timers use the Oracle Advanced Queue to perform queue operations. See the Oracle Advanced Queue documentation set for more information.

## Timer Elements

A timer consists of several mandatory elements, along with as many additional optional elements as necessary to perform the business process. The table below lists these elements and describes them.

***Timer Elements***

| Element | Mandatory | Description |
|---|---|---|
| Message type | yes | Type is timer. |
| Timer name | yes | The display name for the timer. |
| Timer interval | yes | The period of time that the timer is active before it expires. |
| Timer delay | yes | The amount of time to wait before starting the timing period.(The default is zero.) |
| ... | no | Other elements as needed. These elements can be related to a product type, customer category, or Service Level Agreement according to the business requirements of the users. |

> **Note:** All timer delays and intervals use seconds as the unit of measure.

# How Timers Work

After a timer is defined, and linked to an activity (an event), or a message, one of two things can then happen:

- The timer never expires
- The timer expires

### The Timer Never Expires

The following sequence of events occurs if the activity to which a timer is linked completes normally, as scheduled.

1. You use the iMessage Studio to define a timer and associate it with an event, or a message.
2. The Timer Manager enqueues the message in the timer queue (the Oracle Advanced Queue).
3. At run time, the Event Manager treats the timer just like any other message, except that timers are enqueued with a delay and interval which specifies the interval of time after which the message is available for dequeueing.
4. The activity (event or message) to which the timer is linked completes normally.
5. The workflow application deletes any timers related to the response message.
6. Workflow processing continues.

### The Timer Expires

The following sequence of events occurs if the activity to which a timer is linked does not completes normally, and the timer expires.

1. You use the iMessage Studio to define a timer and associate it with an event, or a message.
2. The Timer Manager enqueues the message in the timer queue (the Oracle Advanced Queue).
3. At run time, the Event Manager treats the timer just like any other message, except that timers are enqueued with a delay and interval which specifies the interval of time after which the message is available for dequeueing.
4. The timer interval expires.
5. The Timer Manager dequeues the message from the timer queue and enqueues it on the message queue.
6. Subscribers to this message periodically check the message queue for its presence.
7. Workflow processing continues.

# Message Response Timers versus Window Timers

The Service Delivery Platform uses two types of timers. They are:

- Message response timers
- Window timers

> **Note:** Service Delivery Platform timers use the Oracle Advanced Queue to perform queue operations. See the Oracle Advanced Queue documentation set for more information.

The following table describes each of the two types.

***Service Delivery Platform Timer Types***

| Type | Description |
|---|---|
| Message response timers | These timers are used to respond to messages generated (typically) by workflow activity.<br>The timer interval is used to determine the time allowed to receive a response for the message. |
| Window timers | These timers refer to a specific, well-defined period (or window) of time.<br>For example, you can create a timer to signal the occurrence of some future date minus 24 hours (end date - 24). This is useful if you need to determine whether or not certain events have occurred prior to this time so that the application can continue with its business processes.<br>In this case, the delay would be:<br>delay = (end-date - 24) - current_time |

# Timers and Jeopardy Management

After a timer expires, the message to which it is linked becomes visible in the Oracle Advanced Queue, and the message is published to its subscribers.

This message can be used for a number of purposes, including the following:

- To notify the appropriate personnel to take any necessary action to resolve the jeopardy condition
- To initiate action within a workflow to manage the situation

These kinds of jeopardy management procedures are configurable by the user.

# Jeopardy Notifications

Jeopardy notifications provide messages to Service Delivery Platform users if either of the following occurs:

- A transaction becomes overdue
- A transaction may miss its assigned completion date

## Order Level Jeopardy Notifications

Service Delivery Platform provides a seeded default timer that you can use to generate Order level jeopardy notifications. This timer is known as the Default Jeopardy Timer.

You may use either of the following methods to calculate the timer interval:

- The interval for the timer is calculated based on the Order Due Date (the internal date that stipulates the order completion date). This is the Service Delivery Platform default.
- Alternatively, you may retrieve the timer interval through a stored procedure. Service Delivery Platform provides a stubbed stored procedure for this purpose.

  If you provide the stored procedure, the stubbed procedure can be extended or used to call the user-defined stored procedure. In this case, use the JTF user hooks framework to enable this functionality.

## Configuring a Jeopardy Timer

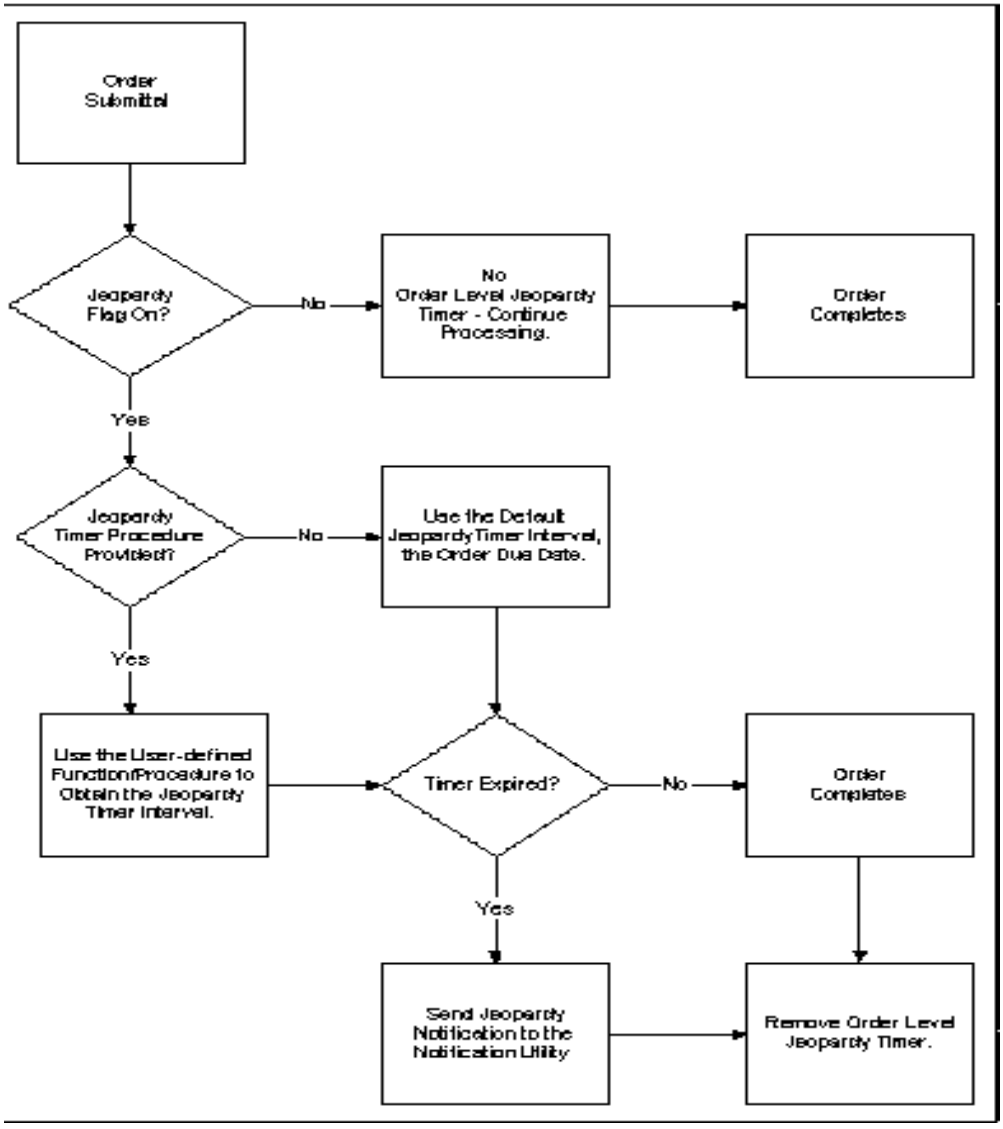
The following steps describe the process of activating a jeopardy timer that is associated with a particular order:

1. You toggle the Jeopardy Enabled Flag in the XDP_OE_Order_Headers by passing it a **Y** or **y** value.
2. The Insert_OE_Order API checks for the value of the Jeopardy Enabled flag. If the value of the flag is **Y** or **y**, then the jeopardy timer is triggered.
3. Upon expiration of the jeopardy timer, the default processing logic for the timer sends a notification to the Notifications utility to indicate that a jeopardy timer for a particular order ID has expired. (Workflow notifications are used to pass the default message to Notifications module.)

4. You remove the jeopardy timer by calling the Remove_Timer API as soon as the order completes. In the call, you must pass the Order_ID and the name of the jeopardy timer to be removed.

The following diagram illustrates the process of configuring a jeopardy notification timer in Service Delivery Platform.

***Jeopardy Notifications in Service Delivery Platform***



# Geographic Areas Defined

Geographic areas are used in Oracle Number Portability for the following purposes:

- To identify the areas covered by a Number Registration Center. The application uses this information to determine to which Number Registration Center a message is to be sent.
- To identify the areas covered by a number range. The geographic area of a number range can be sent to the Number Registration Center whenever a port order is submitted.
- To identify the areas that correspond to a routing number.
- To identify the areas covered by a Service Provider.

## Elements of Geographic Area Definition

You can create new geographic areas or modify an existing area to meet your business needs. Each geographic area contains the following items:

- **Area type:** Geographic area types are user-defined. You must define the area type before beginning to define the geographic area. The application comes with some types already pre-defined.
- **Code:** The code for a geographic area can be defined by a publicly available directory.
- **Name:** The name of the geographic area describes the physical area involved. For example, the name of a city or a region.

**Note:** In defining a geographic area, you can specify that other geographic areas are its children. Thus, if you want to create a hierarchy of geographic areas, then you do this by building the hierarchy from the top downwards starting with the largest geographic areas.

## Service Providers

In order to provide number portability from one service provider to another, each of the service providers must be set up in Oracle SDP Number Portability. This information is used by the Service Order Administrator and the Service Management System to create and activate service orders as they are received.

The precise tasks that you perform for a service provider depend on the purpose of your installation of Oracle SDP Number Portability. The following table lists the three usage types for the application and describes the kind of information that you need for each type.

***Types of Oracle Number Portability Installations***

| Installation Purpose | Tasks |
|---|---|
| Individual Service Provider | You must set up the following:<br>■ Full information for yourself<br>■ Basic information for all other service providers with whom you expect to interact |
| Number Registration Center | You must set up the following:<br>■ Full information for yourself<br>■ Information for all service providers in your region |

## Subscription Versions

Subscription versions are used to maintain the status of porting requests across orders. There are two types of subscription versions. They are:

- **Order Subscription Versions:** Typically created by the recipient service provider, donor service provider, and central system during the Service Order Administration phase of the porting process.

- **Network Subscription Versions:** Typically created by the recipient service provider, donor service provider, and central system during the Service Management System phase of the porting process.

The key components of a Order Subscription Version differ slightly from the key components of a Network Subscription Version.

***Order Subscription Versions versus Network Subscription Versions***

| Component | Order Subscription | Network Subscription |
|---|---|---|
| Phase Indicator | Yes | No |
| Telephone Number | Yes | Yes |
| Routing Number | Yes | Yes |
| Status | Yes | No |
| Porting ID | Yes | Yes |
| Customer | Yes | No |
| Change Cause Code | Yes | No |
| Recipient Service Provider | Yes | No |
| Recipient Service Provider Due Date | Yes | No |
| Donor Service Provider | Yes | No |
| Donor Service Provider Due Date | Yes | No |
| Mediator Service Provider (e.g., Central System) | Yes | Yes |
| Provisioning Map & Provisioning Status | No | Yes |

Oracle SDP Number Portability creates a subscription version for each individual telephone number.

For example, if a range of telephone numbers is ported such as 1234567 through 1234569, the application creates three individual subscription versions, one for each number ported.

## Notes

In general, the following is true about the two types of subscription versions:

1. Participants typically do not have an Order Subscription Version for a given telephone number.

2. Donor and Recipient Service Providers typically have both an Order Subscription Version and a Network Subscription Version for a given telephone number.
3. Central Systems typically have only Order Subscription Versions.

# Order Subscription Versions

An order subscription version is typically created by the recipient service provider, donor service provider, and central system during the Initiation phase of the porting process.

### Recipient Service Provider

After a customer calls a new service provider to request service for his/her existing telephone number, the following events generally occur:

- The recipient service provider creates an order with a porting request for the customer's telephone number.
- This order is passed from the ordering system to the Service Delivery Platform.
- The Service Delivery Platform recognizes the porting request on a specific line item of the order and triggers Oracle SDP Number Portability.
- ONP executes the customized business process for a port-in request and creates a porting order called an Order Subscription Version.
- The recipient operator also sends an outgoing message to notify the central system about the porting request.

The Order Subscription Version is used by the recipient service provider to maintain status of the porting request throughout its life cycle.

### Central System

When the central system receives an incoming message from the recipient operator for a new porting request, the following events generally occur:

- The central system creates a porting order called an Order Subscription Version.
- The central system also sends an outgoing message to notify the donor service provider about the porting request.

The Order Subscription Version is used by the central system to maintain status of the porting request throughout its life cycle.

### Donor Service Provider

When the donor service provider receives an incoming message from the central system for a port-out request for its existing telephone number(s), the following generally occurs:

- The donor service provider creates a porting order called an Order Subscription Version.

The Order Subscription Version is used by the donor service provider to maintain status of the porting request throughout its life cycle.

### Porting ID

The Order Subscription Version for the recipient operator, donor operator and central system is identified by a single porting identifier, the Porting ID number.

- This Porting ID is typically assigned by the central system.
- After assigning a Porting ID, the central system notifies both the recipient operator and the donor operator of the porting ID through messaging.
- A porting ID can reference multiple Subscription Versions. The application does not require a unique porting ID for Subscription Versions.

## Network Subscription Versions

A network subscription version is typically created by the recipient service provider, donor service provider and participant service providers during the Activation phase of the porting process.

### Recipient Service Provider

After a porting request is received, the following events generally occur:

- Prior to the due date of a porting request, the recipient operator must usually make changes to its network to activate service for the new customer.
- At this time, the recipient operator creates a Network Subscription Version using the same Porting ID as the one used to create the Order Subscription Version.

The Network Subscription Version is now used to maintain details and status of each network element that has been updated and the number portability data used to update each network element.

### Donor Service Provider

After a porting request is received, the following events generally occur:

- Prior to the due date of a porting request, the donor operator must also make changes to its network to de-activate service for the new customer and transfer all incoming calls to the recipient service providers network.
- At this time, the donor operator creates a Network Subscription Version using the same Porting ID as the one used to create the Order Subscription Version.

The Network Subscription Version is now used to maintain details and status of each network element that has been updated and the number portability data used to update each network element.

### Participant Service Providers

After a porting request is received, the following events generally occur:

- On the due date or within a specified time period after the porting due date, the participant service providers epochally receives a broadcast from the central system.

  The purpose of the broadcast is to notify all participant service providers that are affected by the customers port to make the necessary changes to their network elements so that all incoming calls to the customer are now delivered to the new service provider instead of the old service provider.

  For example, this broadcast message can contain the Porting ID of the Order Subscription Version created by the central system during the Service Management System phase of the porting process.
- At this time, the participant service providers create a Network Subscription Version using the Porting ID received in the broadcast.

The Network Subscription Version is now used to maintain details and status of each network element that has been updated and the number portability data used to update each network element.

### Porting ID

The Network Subscription Version for the recipient operator, donor operator and participant service providers is uniquely identified by a single porting identifier.

This Porting ID is typically the same Porting ID that was assigned by the central system to uniquely identify the corresponding Order Subscription Version.

# Using Oracle Number Portability

This topic group provides process-oriented, task-based procedures for using the application to perform essential business tasks.

## Finding Information in the NP Center

You use the NP Center to view various kinds of information about orders within the Oracle Number Portability system. This include the following:

- Information about an existing order
- Information about network elements associated with an order

The information displayed in the NP Center is read-only, and can not be modified in any way.

## Viewing Order Information in the NP Center

To view information about an order, perform the following steps.

**Note:** The information displayed in the NP Center is read-only.

### Prerequisites

The order must exist in the system before you can view information about it.

### Steps

1. In the Navigator, choose **Operations > NP Center**.
1. Select the Orders tab.
2. Select an order from the Subscriptions list at the left.

   The Summary screen for that order displays. See Guidelines following for a description of the form fields and their meaning.
3. Click **Details** to display information about a service provider associated with this order.

4. Select the other tabs at the bottom of the screen to display additional information, if desired.
5. Click **Notifications** to open the Notifications Inbox, if desired.

   See Managing Notifications for details of the Notifications Inbox, if necessary.
6. Click the **X** in the upper right-hand corner of the screen to close it, or select Close Form from the File menu.

## Guidelines

The table following lists the kinds of information that is available from the Orders tab of the NP Center.

***Information Available from the Orders Tab***

| Tab | Description |
|---|---|
| Summary | ■ Subscription Summary<br>Customer name and telephone number, order status, routing number, etc.<br>■ Recipient<br>Name and access code of the new service provider. This is Recipient in case of Number Porting, or the new long distance provider in case of Equal Access.<br>■ Donor<br>Name and access code of the old service provider. This is Donor in case of Number Portability or the old long distance provider in case of Equal Access.<br>■ Mediator<br>Name and access code of mediating service provider. This is Number Registration Center in case of Number Portability, or Local Service Provider in case of eqUal Access. |
| Features | ■ The Features tab is not available unless special features have been defined for this order. To enable features, set profile option ENABLE_FEATURES to Y (Yes). |
| Transaction Log | ■ Event Type<br>Type of event that affected the porting record. For example, this could be Status Change or Modification.<br>■ Event<br>Actual event, for example, the status of PENDING.<br>■ Time Stamp<br>Time when the event took place. |

***Information Available from the Orders Tab***

| Tab | Description |
|---|---|
| Work Items | ■ Order ID<br>Service Delivery Platform internal order identification number.<br>■ Order Number<br>External order number as entered in the Order Entry System.<br>■ Work Item<br>Work item number assigned by the system for this order.<br>■ Status<br>State of this work item. It can be received, running, ready, pending, or complete. |
| Failed Downloads | ■ Service Provider Code<br>Code for the Service provider who failed to download the broadcast information<br>■ Service Provider Name<br>Service provider name<br>■ Failure Date<br>Date when the failure took place<br>■ Final Failure Date<br>Date when total failure was declared, meaning that there were no more retries<br>■ Attempted Retries<br>Number of retries attempted to download the information successfully |

***Information Available from the Orders Tab***

| Tab | Description |
|---|---|
| Others | ■ Subscription Owner<br>Service provider code for the owner of the subscription record (in case of Service Bureau)<br>■ Pre-Order Authorization<br>Number which gives the authorization to the service provider to switch the subscriber to another carrier.<br>■ Activation Due Date<br>Date when activation must take place<br>■ Status Change Date<br>Date when the status was last updated<br>■ Invoice Due Date<br>Date when the SP must be billed<br>■ Disconnect Due Date<br>Date when subscriber wishes to disconnect service<br>■ Number Returned Due Date<br>Date when the new Service Provider returns the number to the old Service Provider after disconnecting service<br>■ Effective Release Due Date<br>Date when the subscriber is completely removed from the old SP system<br>■ Donor Cutoff Due Date<br>Date when the donor wants to remove the subscriber from the system |

# Viewing Network Information in the NP Center

To view information about the network elements associated with an order, perform the following steps.

**Note:** The information displayed in the NP Center is read-only, and can not be modified in any way.

## Prerequisites

The order must exist in the system before you can view information about it.

## Steps

1. In the Navigator, choose **Operations > NP Center**.
1. Select the Network tab.
2. Select an order from the Subscriptions list at the left.

   The Summary screen for that order displays. See Guidelines following for a description of the form fields and their meaning.
3. Click **Details** to display information about a service provider associated with this order.
4. Select the other tabs at the bottom of the screen to display additional information, if desired.
5. Click **Notifications** to open the Notifications Inbox, if desired.

   See Managing Notifications for details of the Notifications Inbox, if necessary.
6. Click the **X** in the upper right-hand corner of the screen to close it, or select Close Form from the File menu.

## Guidelines

The table following lists the kinds of information that is available from the Networks tab of the NP Center.

***Information Available from the Network Tab***

| Tab | Description |
|---|---|
| Summary | ■ Subscription summary<br>This includes the customer telephone number, routing Id, etc.<br>■ Routing service provider name<br>Service provider name of the recipient provider (owning the routing number)<br>■ Routing service provider code<br>Service provider code<br>■ Mediating Number Registration Center name<br>Name of the mediating service provider<br>■ Mediating Number Registration Center code<br>Service provider code |

***Information Available from the Network Tab***

| Tab | Description |
|---|---|
| Features | ■ The Features tab will not be available unless special features have been defined for this order. To enable features, set profile option ENABLE_FEATURES to Y (Yes). |
| Work Items | ■ Order Id<br>Order identification number<br>■ Order Number<br>External order number as entered in the Order Entry System<br>■ Work Item<br>Work item number assigned by the system for this order.<br>■ Status<br>State of this work item. (This could be received, running, ready, pending, or complete.) |
| Provisioning Map | ■ Features Type<br>Indicates the feature type that has been provisioned on the network element.For example, this could be: Routing Number, LIDB, or CNAM.<br>(If the Features functionality is not enabled, then only the Routing number for Number Portability is provisioned. This is also called the Primary Routing Number.)<br>■ Network Element<br>Network element name on which the provisioning for this porting record has taken place.<br>■ Network Element Type<br>Network element type on which the provisioning for this porting record has taken place.<br>■ Provision Status<br>Status of the provisioning of the network element. |

# Using the Events Diagnostics Tool

You use the Callback Event Diagnostics page to monitor the runtime activities of the callback registrations used by the Event Manager to route messages to the correct recipients.

## Prerequisites

None

### Steps

1. In the Navigator, choose **Diagnostics > Event**.

   The Callback Event Diagnostics page opens.

2. Enter a value for any of the displayed fields, then click **Find**.

   For example:

   - To search on the order ID, enter a range of values in the Order ID fields and click **Find**.
   - To search on order status, select one from the drop-down list, then click **Find**.
   - To search on a message type, sent to a given Service Provider, enter the Service Provider Code in the Receiver field and click **Find**.

   Any diagnostic messages that meet the search criteria display under Event Details.

3. Close the browser window to exit Diagnostics.

## Using the iMessage Diagnostics Tool

You use the **iMessage Diagnostics** page to search for runtime messages and events based on known details.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Diagnostics > Message**.

   The iMessage Diagnostics page opens.

2. Enter a value for any of the displayed fields, then click **Find**.

   For example:

   - To search for failed messages, enter FAILED in the status field and click **Find**.
   - To search for messages sent to a given Service Provider, enter the Service Provider Code in the Receiver field and click **Find**.

- To search on Code and Event Indicators, click **List** to open a list of the valid indicators, select one, then click **Find**.

Any diagnostic messages that meet the search criteria display under Message Details.

3. Click the Message Id to view the details for a message returned by the search.
4. Click the XML link to view the actual XML version of the message (if your browser supports this option).
5. Close the browser window to exit this diagnostics page.

---

**Note:** If the current message has a status of FAILED, then Message Details page displays a description of the error. A **Fix Message!** link also appears.

Clicking this link re-enqueues a message on the internal events queue for processing.

Although it is possible to correct the error through this link, it is recommend that correct the error through the Notifications Inbox.

---

# Using the Timer Diagnostics Tool

You use the **Timer Diagnostics** page to search for timer messages based on known details. Timers are used by the system for enforcing time-based rules. This could be, for example, jeopardy or service level agreement restrictions.

## Prerequisites

None

## Steps

1. In the Navigator, choose **Diagnostics > Timer**.

   The Timer Diagnostics page opens.

2. Enter a value for any of the displayed fields, then click **Find**.

   For example:

   - 
   - To search on the order Id, enter a range of values in the Order Id fields, and click **Find**.

- To search on a specific Timer type, click **List** to open a list of the valid types, select one, then click **Find**.
- To search on active timers, enter ACTIVE in the status field and click **Find**.

Any diagnostic messages that meet the search criteria display under Message Details.

3. Close the browser window to exit this diagnostics page.

# Using the Configuration Diagnostics Tools

You use the Configuration Diagnostics tool to validate the existing system configuration. The validation process consists of two levels:

**Error:** This type indicates the items that must be corrected for the system to work as expected.

**Warnings:** This type indicates potential problems, but which in some case, are acceptable.

---

**Note:** It is recommended that you run this utility after any major change in the system configuration.

---

## Prerequisites

None

## Steps

1. In the Navigator, choose **Diagnostics > Configuration**.

   The Configuration Diagnostics page opens. This action dynamically executes a validation tool that returns information on the validity of the system configuration.

2. Close the browser window to exit this diagnostics page.

# Managing Geographic Areas

You perform a number of tasks with geographic areas. These include:

- Defining a new geographic area type
- Defining a new geographic area

## Defining a New Geographic Area Type

The Service Delivery Platform comes seeded with certain basic geographic area types. To add new area type, perform the following steps.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Setup > Application Definition > Lookup Codes**

   The Application Object Library Lookups window opens.

2. Enter XNP_AREA_TYPE_CODE in the Type field.

3. Enter the name of the new Lookup Code and its Meaning.

   The Meaning displays when you come to create new geographic areas.

4. Close the window.

   You are prompted to save your changes.

## Defining a Geographic Area

The Service Delivery Platform comes with a seeded WORLD node to which all other geographic areas belong. Do not delete this root node, and do not define any other root node.

To define a new geographic area, perform the following steps.

### Prerequisites

You must define the type to which the new geographic area will belong **before** you define the geographic area itself.

### Steps

1. In the Navigator, choose **Setup > Service Definitions > Geographic Areas**.

2. Click the New icon on the toolbar to create a new geographic area.

3. Enter the name of the new geographic area in the field at the left.

4. Enter its display name, a brief description, and select its area type from the drop-down list.

5. Select the Hierarchy tab if you want to assign child areas to this area, otherwise, skip to step 7.
6. Use the Add / Delete Node icons on the screen to structure the tree hierarchy.
7. Close the window.

   You are prompted to save your changes.

# Defining a Service Provider

To define a service provider, perform the following steps.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Setup > Service Definitions > Service Providers**.
2. Click the New icon on the toolbar to create a new service provider.
3. Enter a name for the new service provider.
4. Use the Details tab to enter basic information about the service provider.
5. Use the Contacts tab to enter contact information for the service provider.

   Note that it is possible to enter more than one contact for a service provider.
6. Use the Numbers tab to enter a number range for each service provider.

   Note that it is possible for one service provider to own a particular number range, but they may have chosen to assign it to another service provider. For the purposes of number portability, ONP always uses the assigned service provider.
7. Use the Filters tab to enter number filters for the service provider.

   A filter is the list of telephone number ranges in which a service provider is interested. You enter this information to create filters so that a Number Registration Center can broadcast your subscription version information selectively.
8. Use the Coverage tab to enter geographic areas supported by a Number Registration Center service provider.

If you are setting up a Number Registration Center as one of the service providers on your system, then you must provide information about those geographic areas supported by this Number Registration Center. You do not have to perform this step for other types of service provider.

- If you do not see the desired area name in the Available box, enter the name in the Filter field and click **Find** to search for it.
- If the area is not defined yet, click **New Geographic Area** to open the Geographic Area Setup window and then add it. See Defining a New Geographic Area for details, if necessary.

9. Use the Routing# tab to enter routing numbers for the service provider.

   Each service provider designates certain number ranges for the purpose of routing ported calls.

   These routing numbers provide the necessary information for routing a ported call. The numbers are mapped to the service provider's network elements and to the ONP subscriber's dialing number.

   Routing numbers are provided by the Number Registration Center for service providers other than you or for those service providers on whose behalf you act. All other service provider's entries are added to and deleted from this table upon instruction from the Number Registration Center.

10. Click the Adapter Configuration tab to enter information about a service provider adapters.
    - Click **New Adapter Configurations** to create a new fulfillment element or to modify an existing fulfillment element, if necessary.

11. Close the window.

    You are prompted to save your changes.

# Defining a Porting Status Type

A service order may have one of a number of different statuses throughout its life cycle. You can define statuses that reflect the terminology and business processes of your organization.

## Prerequisites

Each porting status must be associated with one of the four phases or statuses.

- Inquiry

- Ordering
- Active
- Old

### Steps

1. In the Navigator, choose **Setup > Service Definitions > Porting Status Types**.
1. Click the New icon on the toolbar to create a new porting status type.
2. Enter a value at the left in the Porting Status Code field.
3. Enter a display name and brief description for the new status.
4. Select a phase for the new status.
5. Enter the desired display sequence.
6. Check the Active box to enable the use of this status by Oracle Number Portability.
7. Close the window.

   You are prompted to save your changes.

## Defining Number Ranges and the Network Map

You use the Network Map form window to perform the following two tasks:

- Defining the served number ranges for a given network element
- Defining a network map

## Defining the Served Number Ranges

Perform the following steps to define the served number ranges for a given network element.

### Prerequisites

You must define a fulfillment element before you can associate a range of telephone numbers to it.

### Steps

1. In the Navigator, choose **Setup > Interface Definition > Fulfillment Elements**.

2. Select a fulfillment element from the list at the left.
3. Click **Network Map**.

   The Network Map for <fulfillment element> window opens (where <fulfillment element> is the name that you chose earlier.
4. Select the Served Number Ranges tab.
5. Enter a start and an end number to define the range of telephone numbers with which this fulfillment element is associated.
6. Chose a feature type from the list of values that matches your business process.
7. Select the check box for Primary if the defined number range is to be stored in the primary source for routing information.
8. Select the check box for Active, if the defined number range is to be stored in the active database (as opposed to a backup system).
9. Close the Network Map window.

   You are prompted to save your changes.

# Defining a Network Map

### Prerequisites

None

### Steps

1. In the Navigator, choose **Setup > Interface Definition > Fulfillment Elements**.
2. Select a fulfillment element from the list at the left.
3. Click **Network Map**.

   The Network Map for <fulfillment element> window opens (where <fulfillment element> is the name that you chose earlier.
4. Select the Associated Network Elements tab.

   The information in this tab is for future use and is currently read-only.
5. Close the Network Map window.

# Using the iMessage Studio

You use the iMessage Studio to manage the task of creating new messages. In creating a message, you must perform the following tasks in the order listed.

| Step | Task | See |
|---|---|---|
| 1. | Define the details of the message. | Creating a New Message |
| 2. | Add elements to the message. This includes text, number and date/time fields. | Adding Message Elements |
| 3. | Set the structure of the relationships between the elements of the message. | Setting the Message Structure |
| 4. | Define the source of the information contained in the message. | Defining the Message Data Source |
| 5. | Create any special logic for processing this message. | Defining Your Own Message Processing Logic |
| 6. | Compile the message. | Compiling a Message |
| 7. | Test the message. | Sending a Test Message |

If desired, and it meets your business needs, you can also create a custom notification message.

# Creating a New Message

To create a new message, perform the following steps.

**Warning: Do not create message bodies greater than 32 Kb in length.**

## Prerequisites

None

## Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Click the New icon on the toolbar to create a new message.

3. Enter a name for the message.
4. Select the Details tab.
5. Select Message from the drop-down list of available types.
6. Enter a short name for the message in the Display Name field.

   This name is referenced by the message studio when generating procedures.
7. Enter a brief description of the message.
8. Select a priority for the message.

   This value sets the priority for the message in the outbound or inbound message queue.
9. Select a queue name from the drop-down list.

   If you are uncertain of which to chose, then keep the default value unchanged.
10. Chose a user Responsibility for this message from the drop-down list.
11. Enter the DTD Location path.

    This value sets the path structure to the file that holds all the schema (Document Type Definitions) for this message. The file is named <message>.dtd, where <message> corresponds to the name of the message you are currently defining.
12. Close the window.

    You are prompted to save your changes.

Refer to Adding Message Elements for details of how to enter the information for each message element that makes up this message.

## Adding Message Elements

When creating a message definition, the Message Code and MESSAGE are automatically shown as mandatory messages elements. The message code is defined as the root element.

**Warning: You must never delete the root element, the message code for this message.**

To add a message element to a message, perform the following steps.

### Prerequisites

You must create the message details first, before you can add message elements to it. See Creating a New Message for details.

### Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Select the message to which you wish to add elements from the Message Codes list on the left.
3. Select the Elements tab.
4. Enter a name for the element that you are adding.
5. Enter the datatype for your message element.

   The message element datatype is the XML tag that appears in the message for this element. Supported datatypes are:

   - Text
   - Number
   - Date and time
6. Enter the maximum allowed length of the data.
7. Enter a default value for this message element.
   - You only specify default values for message elements that are parameters.
   - The Send and Publish generated procedures raise an error at run-time if an element does not have a value.
8. Check Mandatory if the message element must be included with the message.
9. Check Parameter if this element is to be used as an argument in generated procedures.

   If a message element is marked as a parameter, then the default value is used. The iMessage Studio generates a CREATE_MSG(), SEND_MSG(), and a PUBLISH_MSG() procedures with the element as a parameter defaulted to the value specified.
10. Enter a value for the sequence order that you wish the parameter to appear in all the generated procedures.
11. Close the window.

    You are prompted to save your changes.

## Setting the Message Structure

The iMessage Studio provides a hierarchical diagramming that you use to define the message structure and to set relationships between message elements. Only predefined elements can be part of this hierarchy.

You can format an XML message by defining its structure here. This allows for complex messages, with master detail relationships.

> **Note:** Elements can be defined more than once within the structure. Also, message elements can be used multiple times throughout the message.

To define the message structure, perform the following steps.

### Prerequisites

You must create the message details first, before you can set the message structure. See Creating a New Message for details.

### Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Select the Structure tab.

   Use the Structure tab to set up the message element hierarchy you require for this message. The onscreen Usage Notes describe how to modify the hierarchy.
3. Close the window.

   You are prompted to save your changes.

## Defining the Message Data Source

To set the source for the information in a message, perform the following steps.

> **Note:** Message elements that are not parameters require a data source.

### Prerequisites

You must create the message details first, before you can define the message source. See Creating a New Message for details.

### Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Select the Data Source tab.
3. Select a message element in the structure tree.
4. Choose a value for Cardinality from the drop-down list, if necessary.

   This values sets whether or not this element is used more than once within this message. You can define the cardinality only if the data source type is set to SQL Query.

   **One or More**: This choice provides the ability to repeat elements within the same message. For example, choose this if you need to set multiple phone numbers for a subscriber.

   **One and Only One**: This choice prevents the message element from repeating itself within the message. If more than one value exists, the first value is used and the other values are ignored. For example, chose this if a request can have only one donor.
5. Select the source type from the drop-down list provided.

   Note that if you chose either PL/SQL Procedure or SQL Query, then you must enter the appropriate code into the Source field also.
6. Enter one of the following in the Reference field:
   a. If the element type is SQL Query, then enter the column number used to provide the value of this element.
   b. If the element type is SDP Parameter, SDP Order Parameter, or SDP FA Parameter, then enter the name of the parameter.
7. Enter one of the following in the Source field:
   a. If the source type is SQL Query, then enter an SQL query. **Do not terminate the SQL query with a semicolon**.
   b. If the source type is PL/SQL Procedure, then enter an SQL function.
8. Close the window.

   You are prompted to save your changes.

# Defining Your Own Message Processing Logic

Incoming messages and events are handled by the Event Manager. There are multiple ways in which a message can be processed. To define the message processing logic, perform the following steps.

## Prerequisites

You must create the message details first, before you can define the message processing logic. See Creating a New Message for details.

## Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Select the Processing Logic tab.
3. Chose a processing logic type from the drop-down list.
   - Default Processing Logic
   - Validate Logic
   - Incoming Message Process Logic
   - Outgoing Message Process Logic

   See the Guidelines following for details of these four choices.
4. Enter the appropriate PL/SQL script.
5. Close the window.

   You are prompted to save your changes.

## Guidelines

The following table lists the four types of message processing logic and provides a brief description of each.

| Type | Description |
|---|---|
| Default Process Logic | If no application has registered for the message, the Event Manager automatically executes the default processing logic DEFAULT_PROCESS() for that message. |
| Validate Logic | The VALIDATE() procedure provides a hook to include business specific validation. It is automatically executed by the Event Manger on the newly arrived message. |

| Type | Description |
|---|---|
| Incoming Message Process Logic | The PROCESS() procedure also provides a hook to include the application logic. It is executed by the Event Manger before delivering the message to the callback procedure of the registered application. |
| Outgoing Message Process Logic | The outgoing process logic is executed before the message is put on the Outgoing Queue for delivery. The user-defined code is executed as part of the SEND() procedure. |

# Compiling iMessages

Perform the following steps to compile a message.

You can also select **Setup > Message Definition > Compile iMessage** to compile a single message, or **Setup > Message Definition > Compile All iMessages** to perform a batch compile.

**Warning: Always ensure that no dequeuers of adapters are running when compiling messages in production.**

## Prerequisites

The message must exist before you can compile it.

## Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Select the Details tab.
3. Select the message you wish to compile from the list at the left.
4. Click **Compile**.
5. Close the window.

   You are prompted to save your changes.

# Sending a Test Message

You test how a message functions in Service Delivery Platform using the iMessage Studio **Test Message** interface. You use this utility to test sending a message to a queue. The message can also be tested using the standard SQL*PLUS interface.

You may use the Order Flowthrough utility, or the Workflow Monitor to monitor the progress of the message.

Errors during processing appear in the Notifications utility. Use this information to correct the error, then resubmit it to the queue for processing, if necessary.

---

**Note:** The Test Message interface can be accessed either from the iMessage Studio or from the Test Center menu.

---

To test a message, perform the following steps.

## Prerequisites

You must first compile the message before you can test it.

## Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Select the Test Message tab.
3. Select the message code from the list at the left for the message you wish to test. The Parameter and Data Type fields fill in automatically.
4. Enter a value in the Reference Id field.
5. This reference value relates the message being sent or received with the current application transaction. For example, this value could be the Order ID.
6. Enter a value in the Opposite Reference Id field.
7. This reference value relates the message being sent or received with the remote application transaction. For example, this value could be the workflow instance.
8. Enter a (comma-separated) list of fulfillment element names in the Consumer List field to which the message or event will be sent (published).
9. Enter the Service Provider Code of the current service provider in the Sender Name field.

   For example, this value could be 9501.

10. Enter the Service Provider Code of the intended recipient service provider in the Recipient List field.

    For example, this value could be 9502. This field may be left empty for events that are being published.

11. Enter the version number of the message being sent.

    This value is incremented only if the original message cannot be processed due to some error.

12. Click **Send** to send the test message.

13. Click **OK** to exit.

    A confirmation window opens containing the Message ID and similar information. It is suggested that you manually record this data as it is useful in tracking a message within the system.

# Working with Event and Timers

You perform a number of tasks relating to events and timers. These include the following:

- Creating a new event
- Creating a new timer
- Associating an event with a subscriber

# Creating a New Event

To create a new event message, perform the following steps.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Click the New icon on the toolbar to create a new event message.
3. Select the Details tab.
4. Select Event from the drop-down list of available types.

5. Enter a short name for the event in the Display Name field.

   This name is referenced by the message studio when generating procedures.

6. Enter a brief description of the event.

7. Select a priority for the event.

   This value sets the priority for the event message in the outbound or inbound message queue.

8. Select a queue name from the drop-down list.

   Select either:

   - Inbound Message Queue
   - Outbound Message Queue

   If you are uncertain of which of these two to chose, then keep the default value unchanged.

9. Chose a user Responsibility for this message from the drop-down list.

10. Enter the DTD Location path.

    This value sets the path structure to the file that holds all the schema (Document Type Definitions) for this message. The file is named <message>.dtd, where <message> corresponds to the name of the message you are currently defining.

11. Close the window.

    You are prompted to save your changes.

## Creating a New Timer

All timers must have a delay and an interval defined as elements. These two elements, Delay and Interval, can be one of the following:

- These elements can be default values.
- These values can be retrieved from a message store using a user-defined procedure.

**Warning: Delay and Interval can not be parameters.**

To create a new timer message, perform the following steps.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Studio**.
2. Click the New icon on the toolbar to create a new timer message.
3. Select the Details tab.
4. Select Timer from the drop-down list of available types.
5. Enter a short name for the event in the Display Name field.

   This name is referenced by the iMessage Studio when generating procedures.
6. Enter a brief description of the timer.
7. Select a priority for the timer.

   This value sets the priority for the timer message in the Timer Message Queue.
8. Select Timer Message Queue from the drop-down list.
9. Chose a user responsibility for this message from the Responsibility drop-down list.
10. Enter the DTD Location path.

    This value sets the path structure to the file that holds all the schema (Document Type Definitions) for this message. The file is named <message>.dtd, where <message> corresponds to the name of the message you are currently defining.
11. Close the window.

    You are prompted to save your changes.

**Note:** You can not add new elements to a timer.

## Associating a Response with an Event

You can associate one or more responses with each defined event. These responses are used by the application workflow in the performance of activities.

**Note:** If a message has an associated timer, the timer must be configured as a response for the event.

To associate a response with an event, perform the following steps.

### Prerequisites

You must first define an event before you can associate a response with it.

### Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Subscribers**.
2. Select the Responses tab.
3. Select an event from the Events pane at the left-hand side of the window.
4. Chose an Event Code from the drop-down list.

   Note that it is possible to link multiple events to the selected event, if desired.
5. Close the window.

   You are prompted to save your changes.

## Registering Default Message Subscribers

To register message subscribers to receive automatic notification when an event occurs, perform the following steps.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Setup > Message Definition > iMessage Subscribers**.
2. Select the Default Subscribers tab.
3. Select an event from the list at the left.
4. Chose a fulfillment element to associate with the event, using the drop-down list.
5. Add additional fulfillment elements, if desired, in the spaces provided.

6. Repeat steps 3 through 5, as many times as necessary.
7. Close the window.

   You are prompted to save your changes.

**Note:** You can also register message subscribers by registering an API using the Callback Registration window.

## Submitting a Test Port In Request

If desired, you can create port in request based on a customer inquiry to test the system. To do so, perform the following steps.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Test Center > Test Port In**.
2. Enter information in the provided fields to create a test port in request.
3. Click **Submit** to save and exit.

# Administering Oracle Number Portability

This topic group provides task-based procedures for required for ongoing system maintenance and includes information on administration tools and utilities.

## System Profile Options

**Note:** To set profile options, you must be logged in as System Administrator, **not** NP System Administrator.

The table following lists the profile options that you can set in the application.

***Service Delivery Platform Profile Options***

| Profile Option Name | Description |
|---|---|
| ACK_REQD_FLAG | Ensures guaranteed delivery of messages. Adapters will send ACK back to calling program.<br>The default is N. |
| DEFAULT_PORTING_STATUS | Initial porting status that is assigned to a porting record. |
| ENABLE_FEATURES | Determine if features such as CNAM, LIDB are available for Installation |
| ENABLE_NRC | Determines if this installation requires use of a central reference database. |
| INSTALL_MODE | Installation mode for operation of Number Portability |
| MAX_RETRIES | The maximum number of attempts that Send Message makes. |
| POP_TIMEOUT | Send Message Pop Time-out |
| SHOW_PROTECTED_MSG | Determines whether iMessage Studio Form displays seed messages |
| SP_NAME | Service Provider Code for Service Bureau mode. |

***Service Delivery Platform Profile Options***

| Profile Option Name | Description |
|---|---|
| TIMER_ACK_TIMEOUT_DURATION | Defines the Time-out duration while waiting for an Acknowledgment messages from the Remote System.<br>Duration is in seconds. |

# The AOL Generic Loader

The Oracle Application Object Library loader is a general purpose data migration tool that is used for patching seed data, delivering translations, or copying setup or transaction data from development to production systems.

The loader is a concurrent program named FNDLOAD. To use this utility, enter the following command at a UNIX prompt.

```
FNDLOAD apps/pwd 0 Y mode configfile datafile entity [ param ... ]
```

The table following lists the parameters used with this executable and describes them.

***FNDLOAD Parameter List***

| Parameter | Description |
|---|---|
| apps/pwd | Specifies the APPS schema and password.<br>■ If the connect_string is omitted, it is taken in a platform-specific manner from the environment using the name TWO_TASK |
| 0 Y | Concurrent program flags |
| mode | Specifies either UPLOAD or DOWNLOAD operation.<br>■ UPLOAD causes the specified data file to be uploaded to the database.<br>■ DOWNLOAD causes the loader to fetch rows and write them to the specified data file. |
| configfile | Specifies the configuration file to use.<br>■ The configuration file usually ends with a suffix of .lct, but this rule is neither enforced nor supplied by the loader. |
| datafile | Specifies the data file to write.<br>■ (DOWNLOAD) If the data file already exists, then it is overwritten.<br>■ The configuration file usually ends with a suffix of .lct, but this rule is neither enforced nor supplied by the loader. |

***FNDLOAD Parameter List***

| Parameter | Description |
|---|---|
| entity | Specifies the entity type to begin the download or upload.<br>■ If you wish to upload all of the entity types in a data file (.ldt), specify a dash (-) as the entity type. |
| param | Specifies zero or more additional parameters that are used to provide bind values in the access SQL (for both the UPLOAD and DOWNLOAD operations).<br>■ Each parameter is of the form NAME=VALUE. The given NAME must not conflict with an attribute name for the entities being loaded. |

## Loader File Definitions

You can find the FNDLOAD configuration files for XNP at the following location:

```
$XNP_TOP/patch/115/import/*.lct
```

The table following lists the Loader files used with Oracle SDP Number Portability and provides the entities and download parameters supported by each. See the contents of the individual configuration file for full documentation on usage.

***Loader File Description (Optional Download Parameters)***

| Name | Description | Entity | Parameters |
|---|---|---|---|
| xnpcbevt.lct | Callback Events | XNP_CALLBACK_EVENTS | MSG_CODE |
| xnpevtsb.lct | Event Subscribers | XNP_EVENT_SUBSCRIBERS | MSG_CODE |
| | | | FE_NAME |
| xnpgeoas.lct | Geographic Areas and Hierarchy | XNP_GEO_AREAS | CODE |
| | | XNP_GEO_HIERARCHY | CODE |
| xnpmsgak.lct | Message Acknowledgments | XNP_MSG_ACKS | MST_CODE |
| xnpmsgps.lct | Messages | XNP_MSG_TYPES | MSG_CODE |
| xnpsnumr.lct | Served Number Ranges | XNP_SERVED_NUM_RANGES | STARTING_ NUMBER |
| | | | FE_NAME |
| | | | FEATURE_TYPE |
| xnpsptre.lct | Service Providers | XNP_SERVICE_PROVIDERS | CODE |
| xnpstats.lct | Porting Status Types | XNP_SV_STATUS_TYPES | STATUS_TYPE_CODE |
| xnptmrpb.lct | Timer Publishers | XNP_TIMER_PUBLISHERS | MSG_CODE |

The following tables lists dependencies between the various files.

***Loader File Dependencies***

| Name | Dependency |
|---|---|
| xnpcbevt.lct | First run xnpmsgps.lct |
| xnpevtsb.lct | First load the fulfillment element data |
| xnpgeoas.lct | First load the Areas, then the Hierarchy |
| xnpmsgak.lct | First run xnpmsgps.lct |
| xnpsnumr.lct | First load the fulfillment element data, then run xnpsptre.lct |
| xnptmrpb.lct | First run xnpmsgps.lct |

### References

For additional information, see the files in the following application directories.

- **Template configuration file**

  /fnddev/fnd/11.5/admin/import/fndstd.lct

- **Existing AOL configuration files**

  /fnddev/fnd/11.5/admin/import/*.lct

## Transferring Lookups to Workflow

To transfer common lookup codes from the configuration tables to workflow, perform the following steps.

### Prerequisites

None

### Steps

1. In the Navigator, choose **Setup > Application Definition > Transfer Lookups to Workflow**.

   The Load NP and OP Lookups onto Workflow window opens with the correct default values already set.

2. Click **Submit** to accept the defaults and start the transfer process.

3. Close this window when the process is complete.

# Downloading Porting Lookups to File

To write the porting lookup codes from workflow into an ASCII file, perform the following steps.

### Prerequisites

You must first transfer the lookups to workflow. See Transferring Lookups to Workflow for details.

### Steps

1. In the Navigator, choose **Setup > Application Definition > Download Porting Lookups to File**.

   The Download NP Lookups from Workflow window opens with the correct default values already set.

2. Click **Submit** to accept the defaults and start the download process.

   This action retrieves the NP lookup codes from the database and stores them in file XNPDNPLK.wft.

3. Close this window when the process is complete.

---

**Note:** To access the output files generated by the Lookup download, refer to the *Oracle Applications 11i System Administrator's Guide*, "Accessing Concurrent Manager Output and Log Files."

---

# Downloading Common Lookups to File

To write the common lookup codes from workflow into an ASCII file, perform the following steps.

### Prerequisites

You must first transfer the lookups to workflow. See Transferring Lookups to Workflow for details.

### Steps

1. In the Navigator, choose **Setup > Application Definition > Download Common Lookups to File**.

The Download OP Lookups from Workflow window opens with the correct default values already set.

2. Click **Submit** to accept the defaults and start the download process.

   This action retrieves the Service Delivery Platform lookup codes from the database and stores them in file XNPDOPLK.wft and XNPDNPLK.wft.

   The lookup codes in the generated files contain the Service Delivery Platform configuration information, including message definitions, fulfillment actions, and similar information. You can use this information for building your business processes.

3. Close this window when the process is complete.

---

**Note:** To access the output files generated by the Lookup download, refer to the *Oracle Applications 11i System Administrator's Guide*, "Accessing Concurrent Manager Output and Log Files."

---

# Creating a Custom Notification Message

Perform the steps following to define a customized notification message that can be displayed at runtime.

### Prerequisites

None

### Steps

1. Create an ASCII text file as a container for the desired message.
2. Name the file with the following format:

   X%_NOTFN_%

   Do not exceed 29 characters total for the length of the file name.

3. Enter the text of your message.

   For example, this could be similar to the following message:

   Porting requested for &STARTING_NUMBER through &ENDING_NUMBER on &NEW_SP_DUE_DATE.

The tokens after the ampersand (&) character are names of work item parameters. At runtime, the notifications utility scans the message and replaces the tokens with the values of the work item parameters.

4. Run the lookup loader script to load these user defined messages onto the workflow lookup code CUSTOMIZED_NOTN_MESSAGES.

The message utility sets the first line of the notification message to item attribute MSG_SUBJECT, and the entire contents of the file to item attribute MSG_BODY. The created notification contains the message with all the referenced work item parameters replaced with the actual value.

# Message Processing Logic in Oracle SDP Number Portability

Incoming messages and events are handled by the Event Manager in Oracle SDP Number Portability. There are four possible ways that a message can be processed by the application.

They are:

- Default process logic
- Validate logic
- Incoming Message process logic
- Outgoing Message process logic

Following is a brief description of each type.

**Note:** If the user does not provide any message processing logic, the default is a NULL package body.

## Default Process Logic

If no application has registered for the message, the Event Manager automatically executes the default processing logic DEFAULT_PROCESS() for that message.

The following example shows how to provide an application hook using the DEFAULT_PROCESS() procedure. Consider a case in which a PORTING_CONCUR message comes in asynchronously.

The default processing logic for this message is:

```
DECLARE
    l_telephone_num
VARCHAR2(10) ;
   l_clli
VARCHAR2(20) ;
    l_area_code
VARCHAR2(3) ;

BEGIN
     /* Reset error code and error message */

     x_error_code := 0 ;
     x_error_message := NULL ;

      /*
      Retrieve the Telephone number message element from the XML message
     */

     XNP_XML_UTILS.DECODE( p_msg_text,

'TELEPHONEí',
                                                                        l_
telephone_num)  ;

     /*
     Retrieve the central office or the CLLI on which it has to be provisioned
     */
     XNP_XML_UTILS.DECODE(  p_msg_text,
                                                                      'CLLI'
,
                                                                       l_
clli) ;

     /*
     Ensure that the right central office is used for provisioning
     */

     l_area_code := SUBSTR(l_telephone_num,1,3 ) ;

     IF ((l_area_code = ë415í) AND
            (l_clli = ëSFOí)) THEN

     /* Customized procedure to provision the number */
     /* Not part of NP core functions */
```

```
          PROVISION.ADD(l_telephone_num) ;
     ELSE
     /*
     Customized procedure to notify the customer care system. Not part of NP
core functions
     */

     NOTIFY_CUSTOMER_CARE(l_tn,
                                                                        x_
error_code,
                                                                        x_
error_message) ;
     END IF ;
END ;
```

### Validate Logic

The VALIDATE() procedure provides a hook to include business specific validation and is automatically executed by the Event Manager on newly arrived messages.

If no validation logic is specified, the procedure is created with a "NULL;" statement. The signature for this procedure is given in the following code.

> **Note:** The Event Manager will not process and deliver the message in case an error is returned in X_ERROR_CODE or X_ERROR_MESSAGE.
>
> However the resulting error code and error message is logged into the system log messages.

```
VALIDATE(
p_msg_header IN XNP_MESSAGE.MSG_HEADER_REC_TYPE,
p_msg_text IN VARCHAR2,
x_error_code OUT NUMBER,
x_error_message OUT VARCHAR2,
p_process_reference IN VARCHAR2 DEFAULT NULL ) ;
```

The following is an example of code that checks for a valid telephone number. (The use of XNP_XML_UTILS.DECODE works only in case of messages with no repeating elements.)

```
DECLARE
     l_telephone_num                                   VARCHAR2(10) ;
```

```
     l_service_provider                                    VARCHAR2(10) ;

BEGIN

     /* Reset error code and error message */

     x_error_code := 0 ;
     x_error_message := NULL ;

      /* Retrieve the telephone number */

     XNP_XML_UTILS.DECODE( p_msg_text,

'TELEPHONE' ,
                                                                        l_
telephone_num ) ;

     /* Retrieve the service provider */

     XNP_XML_UTILS.DECODE(  p_msg_text,

'SERVICE_PROVIDER' ,
                                                                        l_
service_provider )  ;
     /*
     Ccustom procedure to check if the telephone number is in the service
providerís defined number range
     */

     TN_RANGE.CHECK_SP_VALIDITY(  l_telephone_num,

l_service_provider,

x_error_code,

x_error_message ) ;

END ;
```

### Incoming Message Process Logic

The PROCESS() procedure also provides a hook to include the application logic and is executed by the Event manager before delivering the message to the callback procedure of the registered application.

The following example code stores the PORTING_ID from an NPR_ACK for the recipient.

```
DECLARE
 l_REFERENCE_ID VARCHAR2(40) := NULL;
 l_porting_id VARCHAR2(40);
BEGIN

 x_ERROR_CODE := 0;

 /*
  * Get the OPP_REFERENCE_ID as in the received message
  * which is the workitem instance id.
  */
 XNP_XML_UTILS.DECODE
  (p_msg_text
  , 'OPP_REFERENCE_ID'
  , l_REFERENCE_ID
  ) ;

-- Get the NPR PORTING_ID
 XNP_XML_UTILS.DECODE
  (p_msg_text
  , 'PORTING_ID'
  , l_porting_id
  ) ;

  XDP_ENGINE.SET_WORKITEM_PARAM_VALUE
   (to_number(l_REFERENCE_ID)
   ,'PORTING_ID'
   ,l_porting_id
   ,NULL
   );

/* Set the reference to communicate with Number Registration Center, is the
PORTING_ID
*/
  XDP_ENGINE.SET_WORKITEM_PARAM_VALUE
   (to_number(l_REFERENCE_ID)
   ,'OPP_REFERENCE_ID'
   ,l_porting_id
   ,NULL
   );

END;
```

### Outgoing Message Process Logic

The out process logic is executed before enqueueing the message for delivery No procedure is generated but the defined code is executed as part of the SEND() procedure Logging an out going message can be a good use of this hook

The following example code copies a message from the outbound to the inbound queue.

```
BEGIN
DECLARE
  my_header XNP_MESSAGE.MSG_HEADER_REC_TYPE ;
  my_xml  VARCHAR2(4000);
  BEGIN
  my_header := l_msg_header ;
  my_xml := l_msg_text ;
  XNP_MESSAGE.GET_SEQUENCE(my_header.message_id) ;
  my_header.direction_indr := 'I' ;
  XNP_MESSAGE.PUSH(p_msg_header=>my_header,
    p_body_text => my_xml,
    p_queue_name=>XNP_EVENT.C_INBOUND_MSG_Q,
    p_correlation_id=>'MSG_SERVER') ;
  END;
END;
```