Copyright

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.
## Contents

<table>
<thead>
<tr>
<th>About This Document</th>
<th>v</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document Conventions</td>
<td>vi</td>
</tr>
</tbody>
</table>

### Chapter 1

**System Overview** .................................................................................. 1

- Overview................................................................................................. 1
- What is Open Services Development?..................................................... 1
- Typical OSD Configuration Scenario..................................................... 2
- Service Handlers ..................................................................................... 11

### Chapter 2

**Getting Started** .................................................................................. 13

- Overview................................................................................................. 13
- Signing on to Open Services Development.............................................. 13

### Chapter 3

**Using Open Services Development Screens** ...................................... 15

- Overview................................................................................................. 15
- Introduction............................................................................................. 15
- Find Screens............................................................................................ 15
- Service Providers..................................................................................... 16
- Operation Sets......................................................................................... 19
- Operations............................................................................................... 22
- Client ASPs ............................................................................................. 23

### Chapter 4

**Configuration** ..................................................................................... 27

- Overview................................................................................................. 27
- eserv.config Configuration.................................................................... 27
- acs.conf configuration.......................................................................... 38
- SLEE.config Configuration................................................................... 39
- sms.jnlp Configuration......................................................................... 39

### Chapter 5

**Background Processes** ...................................................................... 41

- Overview................................................................................................. 41
- osdlInterface......................................................................................... 41
- WSDL Generating Plug-in....................................................................... 43
- WSDL Regenerator.................................................................................. 44
- Statistics Logged.................................................................................... 44
- Reports.................................................................................................... 44

### Chapter 6

**Troubleshooting** ............................................................................. 49

- Overview................................................................................................. 49
Chapter 7

About Installation and Removal .................................................. 53
Overview ....................................................................................... 53
Installation and Removal Overview ............................................. 53
Post Install Replication ................................................................. 54

NCC Glossary of Terms.................................................................. 55

Index ............................................................................................ 59
About This Document

Scope

The scope of this document includes all functionality a user must know in order to effectively operate the Open Services Development application. It does not include detailed design of the service.

Audience

This guide is written primarily for Open Services Development (OSD) administrators. However, the overview sections of the document are useful to anyone requiring an introduction.

Prerequisites

A solid understanding of UNIX and a familiarity with IN concepts are an essential prerequisite for safely using the information contained in this technical guide. Attempting to install, remove, configure or otherwise alter the described system without the appropriate background skills, could cause damage to the system; including temporary or permanent incorrect operation, loss of service, and may render your system beyond recovery.

Although it is not a prerequisite to using this guide, familiarity with the target platform would be an advantage.

This manual describes system tasks that should only be carried out by suitably trained operators.

Related documents

The following documents are related to this document:

- ACS Technical Guide
- CPE User's Guide
## Document Conventions

### Typographical Conventions

The following terms and typographical conventions are used in the Oracle Communications Network Charging and Control (NCC) documentation.

<table>
<thead>
<tr>
<th>Formatting convention</th>
<th>Type of information</th>
</tr>
</thead>
</table>
| **Special Bold**      | Items you must select, such as names of tabs.  
Names of database tables and fields. |
| **Italics**           | Name of a document, chapter, topic or other publication.  
Emphasis within text. |
| **Button**            | The name of a button to click or a key to press.  
**Example:** To close the window, either click **Close**, or press **Esc**. |
| **Key+Key**           | Key combinations for which the user must press and hold down one key and then press another.  
**Example:** **Ctrl+P**, or **Alt+F4**. |
| **Monospace**         | Examples of code or standard output. |
| **Monospace Bold**    | Text that you must enter. |
| **variable**          | Used to indicate variables or text that should be replaced. |
| **menu option > menu option >** | Used to indicate the cascading menu option to be selected, or the location path of a file.  
**Example:** **Operator Functions > Report Functions**  
**Example:** `/IN/html/SMS/Helptext/` |
| **hypertext link**    | Used to indicate a hypertext link on an HTML page. |

Specialized terms and acronyms are defined in the **Glossary** at the end of this guide.
Overview

Introduction

This chapter provides a high-level overview of the application. It explains the basic functionality of the system and lists the main components.

It is not intended to advise on any specific Oracle Communications Network Charging and Control (NCC) network or service implications of the product.

In this chapter

This chapter contains the following topics.

What is Open Services Development? 1
Typical OSD Configuration Scenario 2
Service Handlers 11

What is Open Services Development?

Introduction

Open Services Development (OSD) enables third parties to submit Web Services Description Language (WSDL) files that invoke control plans.
Architecture

This diagram shows the OSD architecture within a Oracle Communications Network Charging and Control environment.

Typical OSD Configuration Scenario

Introduction

This example is intended to show the nature of completing an OSD configuration, rather than an exact configuration.

OSD configuration process

Configuration for OSD is an iterative process of:

1. Complete Service Provider tab.
2. Complete Operation Sets tab.
3. Complete Operation tab.
4 Complete Client ASP tab.
5 Create control plan for each operation.
6 Compile control plans.
7 Review WSDL through Operation screen (for each control plan).
8 Review WSDL through Operation Sets screen for the service supplier.

Example scenario

This scenario uses a control plan that copies data from the incoming Simple Object Access Protocol (SOAP) request to the outgoing SOAP result (this is not very useful but is a good illustration of how the software works.):

- This copies from a short integer tag called ShortLO profile field in the incoming session data (which is mapped from the) SOAP request to a tag called ShortOPLO in the outgoing session data (which is mapped to the SOAP result).
- The incoming data is the number 12 in this scenario (see In-coming SOAP message).
- The data is copied to ShortOPLO and the response returned through an out-going SOAP message.

To achieve this:
1 The OSD screens are configured as shown.
2 The control plan is created and compiled.
3 The generated WSDL is given to the third party.
4 The third party uses the WSDL by adding the relevant data, and sending the SOAP message to the OSD interface.
5 The OSD interface recognizes what control plan to invoke (see Incoming SOAP message (on page 11)).
6 The control plan copies the number and responds to the OSD interface.
7 The OSD interface returns a SOAP message to the third party (see Outgoing SOAP message (on page 11)).

Service Providers tab

Here is the Service Providers tab, configured for the scenario.

Notes:
- The UAS port and address configured below must match the VWS eserv.config file triggering section address and port information.
The InterfaceName must match the configured OSD interface running on the SLC SLEE.

**Operation Sets tab**

Here is the Operation Sets tab, configured for the scenario.

**Note:** The service to invoke is the capability that will be triggered on the SLC SLEE. For example, if the CCS service loader is used, this field will specify the CCS capability that will be triggered.

**Operations tab**

Here is the Operations tab, configured for the scenario.
**Note:** The control plan name is populated automatically when this operation is selected for WSDL generation during control plan compilation. The operation should only be enabled after a control plan WSDL has been generated.

**Client ASPs tab**

Here is the Client ASPs tab, configured for the scenario.

**Notes:**

- The client ASP name and IP address are the details of the VWS.
The user name and password information configured must match the triggering operations overrides section of the `eserv.config` file on the VWS.
Control Plan

Here is the control plan for the scenario.

Important: An OSD control plan must contain an Unconditional Terminate or Unconditional Terminate to Pending feature node.
Copy Node Configuration

Here is the Copy node configuration detail for this scenario.

Control plan compilation

When the control plan is compiled, parameters are inserted into the WSDL using the OSD and Copy node configuration screens to provide the parameter names:

- "CC_Service_Number" - is the number to trigger the control plan which we configured in ACS-number screen
- "ShortLO" - is the input which we have to provide in the SOAP message
- "ShortOPLO" - is the output which we have to provide in the SOAP message
- "errorCode" - is for error messages, so if the functionality is not correct or we have wrong SOAP message the error code will come out in outgoing SOAP messages

The resultant operation segment is generated in three parts:

- soakRequestType
- soakResultType
- soakFaultType

```
<x:s:complexType name="soakRequestType">
  <xs:sequence>
    <xs:element name="CC_Service_Number" type="OCNCC:NumericString" minOccurs="1"/>
    <xs:element name="ShortLO" type="OCNCC:Short" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<x:s:complexType name="soakResultType">
  <xs:sequence>
    <xs:element name="ShortOPLO" type="OCNCC:Short" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<x:s:complexType name="soakFaultType">
  <xs:sequence>
    <xs:element name="errorCode" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```
Operation WSDL

This is the operation generated WSDL code from the control plan compilation.

```
<xsd:element name="soakRequest" type="soakRequestType"/>
<xsd:element name="soakResult" type="soakResultType"/>
<xsd:element name="soakFault" type="soakFaultType"/>
<xsd:complexType name="soakRequestType">
  <xsd:sequence>
    <xsd:element name="CC_Service_Number" type="OCNCC:NumericString" minOccurs="1"/>
    <xsd:element name="ShortLO" type="OCNCC:Short" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="soakResultType">
  <xsd:sequence>
    <xsd:element name="ShortOPLO" type="OCNCC:Short" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="soakFaultType">
  <xsd:sequence>
    <xsd:element name="errorCode" type="xs:int"/>
  </xsd:sequence>
</xsd:complexType>
```

Note: The "===>" lines are not part of the generated file, they mark the start and end lines of the operation WSDL lines.

Operation Set WSDL

This is the full WSDL generated for the operation set.
<?xml version="1.0"?>
<definitions name="soak"
    targetNamespace="http://boss-sb-smp/wsdls/OSD/soak.wsdl"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:tns="http://boss-sb-smp/wsdls/OSD/soak.wsdl"
    xmlns="http://schemas.xmlsoap.org/wsdl/">
    <types>
        <xs:element name="soakRequest" type="soakRequestType"/>
        <xs:element name="soakResult" type="soakResultType"/>
        <xs:element name="soakFault" type="soakFaultType"/>
        <xs:complexType name="soakRequestType">
            <xs:sequence>
                <xs:element name="CC_Service_Number" type="OCNCC:NumericString" minOccurs="1"/>
                <xs:element name="ShortLO" type="OCNCC:Short" minOccurs="0"/>
            </xs:sequence>
        </xs:complexType>
        <xs:complexType name="soakResultType">
            <xs:sequence>
                <xs:element name="ShortOPLO" type="OCNCC:Short" minOccurs="0"/>
            </xs:sequence>
        </xs:complexType>
        <xs:complexType name="soakFaultType">
            <xs:sequence>
                <xs:element name="errorCode" type="xs:int"/>
            </xs:sequence>
        </xs:complexType>
    </types>
    <message name="soakInput">
        <part name="body" element="tns:soakRequest"/>
    </message>
    <message name="soakOutput">
        <part name="body" element="tns:soakResult"/>
    </message>
    <message name="soakFaultOutput">
        <part name="body" element="tns:soakFault"/>
    </message>
    <portType name="soakPortType">
        <operation name="soakOperation">
            <input message="tns:soakInput"/>
            <output message="tns:soakOutput"/>
            <fault message="tns:soakFaultOutput"/>
        </operation>
    </portType>
    <binding name="soakBinding" type="tns:soakPortType">
        <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
        <operation name="soakOperation">
            <soap:operation soapAction="http://boss-sb-smp/wsdls/OSD/soak/soak"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
            <fault>
                <soap:body use="literal"/>
            </fault>
        </operation>
    </binding>
</definitions>
Incoming SOAP message

This is the incoming SOAP message used to trigger the control plan.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:tags="http://abox-sb-smp/wsdls/OSD/tags.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <tags:acsTagsRequest xmlns="http://abox-sb-smp/wsdls/OSD/tags.wsdl">
      <!--Optional:-->
      <Short>12</Short>
      <CC_Service_Number>1234567</CC_Service_Number>
    </tags:acsTagsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Outgoing SOAP message

This is the outgoing SOAP message.

```xml
<?xml version="1.0"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding">
  <SOAP-ENV:Body>
    <m:reTestResult xmlns:m="http://abox-sb-smp/wsdls/OSD/reTest.wsdl">
      <ShortOPLO>12</ShortOPLO>
    </m:reTestResult>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Service Handlers

Introduction

These service handlers are available for selection in the Service to Invoke drop down list, see Operations (on page 22). The list is governed by what software is installed on the platform.

If both OSD and CCS are installed the two lists will be combined.

OSD service handlers

These are the possible services to invoke that are installed by OSD.

- ACS
- ACS_Management
- ACS_OSD
- ACS_Outgoing
- ACS_Prefix

CCS service handlers

These are the possible services to invoke that are installed when Prepaid Charging is also installed on the platform.

- CCS
- CCS_ROAM
- CCS_SM_MO
Chapter 1

- CCS_SM_MT
- REVERSE_CCS_SM_MT
- CCS_DATA
- CCS_MO
- CCS_MT
- CCS_BPL
Overview

Introduction
This chapter explains how to start the Open Services Development screens.

In this chapter

This chapter contains the following topics.

Signing on to Open Services Development

Signing on to Open Services Development

Overview

To access Open Services Development (OSD), you need to log into Service Management System and select Open Services Development from the Services menu on the SMS main screen. Refer to SMS User's Guide.

SMS main screen

Here is an example of the Service Management System main menu showing the Open Services Development menu option.
Overview

Introduction

This chapter explains what and how to use the OSD configuration screens.

In this chapter

This chapter contains the following topics.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>15</td>
</tr>
<tr>
<td>Find Screens</td>
<td>15</td>
</tr>
<tr>
<td>Service Providers</td>
<td>16</td>
</tr>
<tr>
<td>Operation Sets</td>
<td>19</td>
</tr>
<tr>
<td>Operations</td>
<td>22</td>
</tr>
<tr>
<td>Client ASPs</td>
<td>23</td>
</tr>
</tbody>
</table>

Introduction

The OSD configuration screens allow for setting up how to handle incoming SOAP messages, for any OSD data specific to a service provider, that is, one ACS customer.

Find Screens

Introduction

Find is available of the Operation Sets, Operation and Client ASPs tabs.

While each of these tabs has a different results table, they all use the same mechanism to populate their tables.
Example Find screen

Here is an example find screen with the default results table for the selected service provider.

![Find Operations Set](image)

Using the Find screen

If the Service Provider drop down list is present, it can be ignored, unless you are wanting to change provider on all the tabs.

The search text box (Operation Set Name in the example) is used to do a search on items beginning with the text typed. For example typing "Weekly" would return all operation sets whose name started with "Weekly".

**Note:** This search function is case sensitive - "week" would not find anything.

To start the search, click Search. All found items replace any previous table contents.

Once the record has been found, click on that table entry and click Close to return to the parent tab, which will be populated with the found record details.

Service Providers

Introduction

The Service Providers tab allows you to select a service provider that will have OSD configuration.

**Note:** Once selected, data for the service provider is propagated to the following tabs:

- Service Providers
- Operation Sets
- Operations
Service Providers tab

Here is an example of the Service Providers tab.

![Image of Service Providers tab]

Service Providers fields

This table describes the Service Provider tab fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Provider</td>
<td>The service provider that an ASP uses for the operation set.</td>
</tr>
<tr>
<td>Note:</td>
<td>This will be the same provider for the Service Provider, Operation Sets and Operation tabs.</td>
</tr>
<tr>
<td></td>
<td>Selection of a different provider changes the screen contents as if Clear had been clicked.</td>
</tr>
<tr>
<td>Use Router</td>
<td>Flag to indicate that ASPs using operations belonging to this service provider access OSD through a router such as squid.</td>
</tr>
<tr>
<td></td>
<td>If this flag is selected then the router port and router address are placed in the WSDL file.</td>
</tr>
<tr>
<td></td>
<td>If this flag is not selected then ASPs access OSD on the SLCs directly for these operations. In this case, all the ports and addresses from the SLC ports panel are placed in the WSDL file and the router port and router address are not used.</td>
</tr>
</tbody>
</table>
Edit Service Providers

Follow these steps to edit a service provider OSD interaction.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
</table>
| 1    | Select the **Service Provider** from the drop down list.  
**Note:**  
- This is a list of already established service providers (see **SMS Main menu > Services > ACS Services > Customers** tab.  
- The selected service provider is auto selected in the other tabs. |
| 2    | If load sharing is required, select **Use Router** check box, then:  
- Enter the router port in the **Router Port** field  
- Enter the router address in the **Router Address** field |
| 3    | Select the **Protocol** to be used:  
- HTTP  
- HTTPS |
| 4    | Amend the list of SLC ports to receive ASP input from.  
- To add a new port - see **Adding SLC ports** (on page 18)  
- To change a port - see **Editing SLC ports** (on page 19)  
- Select a port and click **Remove** to delete a port from the list  
Then confirm on the Delete Port Entry confirmation dial. |
| 5    | Click **Save**. |

Adding SLC ports

Follow these steps to add a SLC port.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
</table>
| 1    | Click **Add**.  
**Result:** The New Port List Entry screen appears. |
| 2    | Enter the new port number in the **Port** field. Must be 1024 through 65535. |
| 3    | Enter the port address in the **Address** field. Must be the host name or IP address of a |
Chapter 3

Step | Action
--- | ---
4 | SLC running OSD.

Enter the interface name in the **Interface Name** field. Must match the name of a running osdInterface on the SLC, as defined in the **SLEE.cfg** file.

More than one interface may be used for performance reasons.

5 | Click **Save**.

**Editing SLC ports**

Follow these steps to edit an existing SLC port list entry.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
</table>
1 | Select the port entry to change from the table. |
2 | Click **Edit**.  
**Result:** The Edit Port List Entry screen appears. |
3 | Enter the port address in the **Address** field. Must be the host name or IP address of a SLC running OSD. |
4 | Type the interface name in the **Interface Name** field. Must match the name of a running osdInterface on the SLC, as defined in the **SLEE.cfg** file.  
More than one interface may be used for performance reasons. |
5 | Click **Save**.

**Operation Sets**

**Introduction**

Operation sets are a collection, for ease of maintainability, of related operations.

The **Operation Sets** tab is where the selected service provider has all their sets of operations configured. Each set can have any number of operations (see **Operations** on page 22) and each service provider can have any number of operation sets.

When generated by a control plan compile, all operations for the operation set are inserted into a single WSDL file.
Operation Sets tab

Here is an example of the Operation Sets tab.

![Operation Sets Tab](image)

Operation Sets fields

This table describes the Operation Sets tab fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Provider</td>
<td>The service provider for this operation set.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong></td>
</tr>
<tr>
<td></td>
<td>• This will be the same provider for the Service Provider, Operation Sets</td>
</tr>
<tr>
<td></td>
<td>and Operation tabs.</td>
</tr>
<tr>
<td></td>
<td>• Selection of a different provider changes the screen contents as if</td>
</tr>
<tr>
<td></td>
<td>Clear had been clicked.</td>
</tr>
<tr>
<td>Operation Set Name</td>
<td>The name of this collection of operations.</td>
</tr>
<tr>
<td>WSDL Location</td>
<td>Location of the control plan generated WSDL file for this operation set.</td>
</tr>
<tr>
<td>WSDL URL</td>
<td>Web URL for the WSDL file.</td>
</tr>
</tbody>
</table>

Note: This is set at installation time in the `sms.jnlp` file and in `eserv.config`. 
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>wsdlUriBaseName parameter.</td>
<td></td>
</tr>
<tr>
<td>Service to Invoke</td>
<td>The service that will be used to invoke the control plan from the WSDL file. This must match a configured service on the SLCto trigger the control plan successfully.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong></td>
</tr>
<tr>
<td></td>
<td>• This list is created at SMS package installation time in a database table. See Service Handlers.</td>
</tr>
<tr>
<td></td>
<td>• For meanings and uses of the different service handles, see ACS Technical Guide and CCS Technical Guide.</td>
</tr>
<tr>
<td>Max outstanding Transactions</td>
<td>This is the maximum number of SOAP requests for operations in this operation set that are allowed to be active at any one time. Any SOAP requests in excess of this will be rejected with HTTP error 503 (unavailable).</td>
</tr>
</tbody>
</table>

**Warning:** If either the WSDL location or WSDI URL are modified, then it is up to the user to configure the SMS file system and Apache so that the two are consistent.

### Editing Operation Sets

Follow these steps to edit a service provider operation set.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the Service Provider from the drop down list.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong></td>
</tr>
<tr>
<td></td>
<td>• This is a list of already established service providers (see SMS Main menu &gt; Services &gt; ACS Services &gt; Customers tab.</td>
</tr>
<tr>
<td></td>
<td>• The selected service provider is auto selected in the other tabs.</td>
</tr>
<tr>
<td>2</td>
<td>To set the tab for a new operation set, click <strong>Clear</strong>.</td>
</tr>
<tr>
<td></td>
<td>To locate an existing operation set for amending, click <strong>Find</strong> (see Find screens (on page 15)).</td>
</tr>
<tr>
<td></td>
<td>To remove an operation set, click <strong>Delete</strong>, and then confirm on the Delete Operation Set confirmation dialog.</td>
</tr>
<tr>
<td>3</td>
<td>Type the operation set name in the Operation Set Name field.</td>
</tr>
<tr>
<td></td>
<td><strong>Result:</strong> The name is inserted into the WSDL file location and URL.</td>
</tr>
<tr>
<td>4</td>
<td>Select the Service to Invoke from the drop down list.</td>
</tr>
<tr>
<td></td>
<td><strong>Tip:</strong> This should reflect the service the WSDL is going to invoke.</td>
</tr>
<tr>
<td>5</td>
<td>Type the maximum allowed outstanding transactions in the Max Outstanding Transactions field.</td>
</tr>
<tr>
<td>6</td>
<td>Click <strong>Save</strong>.</td>
</tr>
</tbody>
</table>
Operations

Operations tab

Here is an example of the Operations tab.

![SU - Open Services Development](image)

Operations fields

This table describes the Operations tab fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Provider</td>
<td>The service provider for this operation.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong></td>
</tr>
<tr>
<td></td>
<td>• This will be the same provider for the Service Provider, Operation Sets and Operation tabs.</td>
</tr>
<tr>
<td></td>
<td>• Selection of a different provider changes the screen contents as if Clear had been clicked.</td>
</tr>
<tr>
<td>Operation Name</td>
<td>The name of this operation. This is the name that the control plan uses when generating the WSDL file sub set.</td>
</tr>
<tr>
<td>Operation Set</td>
<td>The operation set that this operation will belong to.</td>
</tr>
<tr>
<td></td>
<td><strong>Tip:</strong> For periodic charges this must be PeriodicCharge.</td>
</tr>
<tr>
<td>Control Plan</td>
<td>The control plan that this WSDL sub set will invoke. This is automatically populated when saving a control plan with this operation name.</td>
</tr>
<tr>
<td>Enabled</td>
<td>If an operation is not enabled, the ASP will receive a SOAP fault with error 7 = operation not available.</td>
</tr>
</tbody>
</table>
Editing Operations

Follow these steps to edit a service provider operation.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the Service Provider from the drop down list.</td>
</tr>
</tbody>
</table>
| 2    | To set the tab for a new operation, click **Clear**.  
To locate an existing operation for amending, click **Find** (see **Find screens** (on page 15)).  
To remove an operation, click **Delete**, then confirm on the Delete confirmation dialog. |
| 3    | Type the operation name in the **Operation Name** field.  
**Result:** The name is inserted into the WSDL file location and URL. |
| 4    | Select the **Operation Set** for this operation from the drop down list. |
| 5    | Select the **Enabled** check box. |
| 6    | Click **Save**. |
| 7    | Repeat steps 4 through 6 for each operation set to add this operation to. |

Client ASPs

Client ASPs tab

Here is an example Client ASPs tab.
Client ASPs fields

This table describes the function of each field.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client ASP Name</td>
<td>The name of the ASP.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IP address for the ASP.</td>
</tr>
<tr>
<td>User Name</td>
<td>The user name for this ASP. The combination of user name and IP address is used to identify the ASP.</td>
</tr>
<tr>
<td>Max Tx/Sec</td>
<td>The maximum number of SOAP requests per second this ASP is allowed to send. Any SOAP requests in excess of this will be rejected with HTTP error 503 (unavailable).</td>
</tr>
<tr>
<td>Max Tx Outstanding</td>
<td>This is the maximum number of SOAP requests from this ASP that are allowed to be active at any one time. Any SOAP requests in excess of this will be rejected with HTTP error 503 (unavailable).</td>
</tr>
<tr>
<td>Change Password</td>
<td>The password used to authenticate this ASP.</td>
</tr>
<tr>
<td>Confirm Password</td>
<td>This must match Change Password value.</td>
</tr>
</tbody>
</table>

Edit Client ASPs

Follow these steps to edit a client ASP.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>To amend or delete an ASP, use the Find functionality to locate the required ASP (see Find screens (on page 15)).</td>
</tr>
<tr>
<td>2</td>
<td>To add a new ASP, click Clear.</td>
</tr>
<tr>
<td>3</td>
<td>Enter the new ASP name in the Client ASP Name field.</td>
</tr>
<tr>
<td>4</td>
<td>Enter or change the IP Address.</td>
</tr>
<tr>
<td>5</td>
<td>Enter or change the User Name.</td>
</tr>
<tr>
<td>6</td>
<td>Enter or change the maximum transaction rate in the Max Tx/Sec field.</td>
</tr>
<tr>
<td>7</td>
<td>Enter or change the maximum transaction backlog in the Max Tx Outstanding field.</td>
</tr>
<tr>
<td>8</td>
<td>Set the password for the SOAP HTML header in the Change Password and Confirm Password fields.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Any password will do, but a secure password containing characters and numbers is recommended.</td>
</tr>
<tr>
<td>9</td>
<td>Click Save.</td>
</tr>
<tr>
<td></td>
<td><strong>Result:</strong> The Add.. becomes available to do the next step.</td>
</tr>
<tr>
<td>10</td>
<td>Maintain the allowed operations for this ASP (see Add Allowed Operations (on page 24)).</td>
</tr>
</tbody>
</table>

Add Allowed Operations

Follow these steps to add an allowed operation.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Click Add.</td>
</tr>
<tr>
<td></td>
<td><strong>Result:</strong> The Allow Operation for ASP screen appears, with the fields defaulted as</td>
</tr>
</tbody>
</table>
Chapter 3

Using Open Services Development Screens

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>If required, select a new <strong>Service Provider</strong> from the drop down list. <strong>Note:</strong> The selected provider is also changed in the other OSD tabs and a prompt is made if there are unsaved changes.</td>
</tr>
<tr>
<td>3</td>
<td>If required, select the <strong>Operation Set</strong> from the drop down list.</td>
</tr>
<tr>
<td>4</td>
<td>If required, select the <strong>Operation</strong> from the drop down list.</td>
</tr>
<tr>
<td>5</td>
<td>Click <strong>Save</strong>. <strong>Result:</strong> The operation is added to the <strong>Allowed Operations</strong> table.</td>
</tr>
</tbody>
</table>

### Remove Allowed Operations

Follow these steps to remove an allowed operation from this ASP.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the allowed operation to remove from this ASP from the <strong>Allowed Operations</strong> table.</td>
</tr>
<tr>
<td>2</td>
<td>Click <strong>Remove</strong>. <strong>Result:</strong> The confirmation screen appears.</td>
</tr>
<tr>
<td>3</td>
<td>Click <strong>Remove</strong> to confirm the removal. <strong>Result:</strong> The allowed operation is removed from the list.</td>
</tr>
</tbody>
</table>
Overview

Introduction

This chapter explains how to configure the Oracle Communications Network Charging and Control (NCC) application.

In this chapter

This chapter contains the following topics.

eserv.config Configuration 27
acs.conf configuration 38
SLEE.config Configuration 39
sms.jnlp Configuration 39

eserv.config Configuration

Introduction

The eserv.config file is a shared configuration file, from which many NCC applications read their configuration. Each NCC machine (SMS, SLC, and VWS) has its own version of this configuration file, containing configuration relevant to that machine. The eserv.config file contains different sections; each application reads the sections of the file that contains data relevant to it.

The eserv.config file is located in the /IN/service_packages/ directory.

The eserv.config file format uses hierarchical groupings, and most applications make use of this to divide up the options into logical groupings.

Configuration file format

To organize the configuration data within the eserv.config file, some sections are nested within other sections. Configuration details are opened and closed using either { } or [ ].

- Groups of parameters are enclosed with curly brackets - { }
- An array of parameters is enclosed in square brackets - []
- Comments are prefaced with a # at the beginning of the line

To list things within a group or an array, elements must be separated by at least one comma or at least one line break. Any of the following formats may be used, as in this example:

```plaintext
{ name="route6", id = 3, prefixes = [ "00000148", "0000473"] }
{ name="route7", id = 4, prefixes = [ "000001049" ] }
```

or

```plaintext
{ name="route6"
  id = 3
  prefixes = [ "00000148"
              "0000473"
             ]
```
eserv.config files delivered

Most applications come with an example eserv.config configuration in a file called eserv.config.example in the root of the application directory, for example, /IN/service_packages/eserv.config.example.

Editing the file

Open the configuration file on your system using a standard text editor. Do not use text editors, such as Microsoft Word, that attach control characters. These can be, for example, Microsoft DOS or Windows line termination characters (for example: ^M), which are not visible to the user, at the end of each row. This will cause file errors when the application tries to read the configuration file.

Always keep a backup of your file before making any changes to it. This will ensure you have a working copy to which you can return.

Loading eserv.config changes

If you change the configuration file, then you must restart the appropriate parts of the service to enable the new options to take effect.

Re-reading the config file

The system will re-read config on a SIGHUP signal and print a status report to standard output on a SIGUSR1 signal.

OSD eserv.config configuration -ncc5.0.0

Here is an example OSD eserv.config section configuration.

OSD = {
    osdInterface = {
        basicRealm = "ASP Account"
        pollTime = 10000
        loadReportingPeriod = 600
        oracleusername = "smf"
        oraclepassword = "smf"
        oracledatabase = "nzwn-test08_SMF"
        sslCertificateFile = "/IN/service_packages/OSD/sslCertificate.pem"
        sslKeyFile = "/IN/service_packages/OSD/sslKey.pem"
        tssfTimeOutSecs = 10
        persistentConnection = true
        connectionTimeOutSecs = 60
        rateCalculationPeriodSecs = 10
    }
}
applicationContext = "0,4,0,0,1,21,3,4"
validateRequestNameSpace = false
databaseCachingRules = {
    operationSetsDataExpirySecs = 300
    operationDataExpirySecs = 300
    clientAspDataExpirySecs = 300
    portListsDataExpirySecs = 300
    acsProfileDataExpirySecs = 300
}
NumberRules = [
    { prefix="", min=0, max = 100, remove=0, prepend="", resultNoa=4 }
]
}
wsdlUriBaseName = "http://nzwn-test08.uk.oracle.com/wsdls"
use DeprecatedSchema = false
maxProfileDetailsAge = 60
maxServiceHandleAge = 60
osdWsdlRegenerator = {
    waitTimeSecs = 2
    useOracleAlerts = false
    oracledatabase = "nzwn-test08_SMF"
}

**OSD SLC parameters**

Here are the parameters in the osdInterface section of the eserv.config OSD configuration.

**Note:** osdInterface parameters are only relevant on an SLC.

**applicationContext**

**Syntax:**

applicationContext = "context"

**Description:**
The application context to specify in IDPs sent to slee_acs.

**Type:**
String

**Optionality:**
Optional (default used if not set).

**Allowed:**

**Default:**
"0,4,0,0,1,21,3,4"

**Notes:**
This should not normally be changed.

**Example:**
applicationContext = "0,4,0,0,1,21,3,4"

**basicRealm**

**Syntax:**

basicRealm = "realm"

**Description:**
The basic realm to specify in HTTP authentication (401) messages, WWW-Authenticate header.

**Type:**
String

**Optionality:**
Optional (default used if not set).

**Allowed:**

**Default:**
"ASP Account"

**Notes:**

**Example:**

basicRealm = "ASP Account"
connectionTimeOutSecs
Syntax: \texttt{connectionTimeOutSecs = seconds}
Description: The http connection timeout in seconds.
Type: Integer, Decimal, Array, Parameter list, String, Boolean
Optionality: Optional (default used if not set).
Allowed: None
Default: \begin{itemize}
  \item 60 for persistent connection mode
  \item 10 for single connection mode
\end{itemize}
Notes: Default depends upon \texttt{persistentConnection} setting.
Example: \texttt{connectionTimeOutSecs = 60}

loadReportingPeriod
Syntax: \texttt{loadReportingPeriod = seconds}
Description: Number of seconds between automatic status reports.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: None
Default: 600
Notes: Usually not specified as it is usual to use the local database.
Example: \texttt{loadReportingPeriod = 900}

oracledatabase
Syntax: \texttt{oracledatabase = "database_name"}
Description: The remote Oracle database to connect to, from tnsnames.ora.
Type: String
Optionality: Optional (default used if not set).
Allowed: None
Default: None
Notes: Usually not specified as it is usual to use the local database.
Example: \texttt{oracledatabase = "nzwn-test08_SMF"}

Syntax: \texttt{oraclepassword = "password"}
Description: Oracle password for connecting to the database.
Type: String
Optionality: Optional.
Allowed: None
Default: None
Notes: Usually the \texttt{oracleusername} and \texttt{oraclepassword} are not specified in which case the \texttt{acs_oper} operator account will just connect as "/".
Example: \texttt{oraclepassword = "smf"}

oracleusername
Syntax: \texttt{oracleusername = "user_name"}
Description: Oracle user name for connecting to the database.
Type: String
Optionality: Optional.
Allowed:
Default:
Notes: Usually the oracleusername and oraclepassword are not specified in which case the acs_oper operator account will just connect as "/".
Example: oracleusername = "smf"

osdInterface
Syntax: osdInterface = {parm1, parm2...}
Description: The OSD interface parameter list.
Type: List
Optionality: Mandatory
Allowed:
Default: None
Notes: osdInterface parameters are only relevant on an SLC.
Example:
   osdInterface = {
       parm 1
       parm 2
   }

persistentConnection
Syntax: persistentConnection = true|false
Description: Run in persistent http connection mode.
Type: Boolean
Optionality: Optional (default used if not set).
Allowed:
   - true - run in persistent http connection mode
   - false - run in single http connection mode
Default: true
Notes:
Example: parameter = persistentConnection = true

pollTime
Syntax: pollTime = microseconds
Description: Number of microseconds to wait for a socket connection or SOAP request.
Type: Integer
Optionality: Optional (default used if not set).
Allowed:
Default: 10000
Notes:
Example: pollTime = 20000

rateCalculationPeriodSecs
Syntax: rateCalculationPeriodSecs = seconds
Description: The number of seconds for which to store transactions when calculating the transaction rate.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: 10
Default: 10
Notes: For example with the default of 10 seconds, and with an allowed rate of 200 per second, new transactions will be rejected if the number of transactions started in the last 10 seconds exceeds 2000.
Example: rateCalculationPeriodSecs = 20

sslCertificateFile
Syntax: sslCertificateFile = "location/file_name"
Description: Name and location of the SSL certificate file.
Type: String
Optionality: Optional (default used if not set).
Allowed: 
Default: "/IN/service_packages/OSD/sslCertificate.pem"
Notes: The sslCertificate file must be created manually in order for SSL to work.
Example: sslCertificateFile = "/IN/service_packages/OSD/sslCertificate.pem"

sslKeyFile
Syntax: sslKeyFile = "location/name"
Description: Name and location of the SSL key file for certificate.
Type: String
Optionality: Optional (default used if not set).
Allowed: 
Default: "/IN/service_packages/OSD/sslKey.pem"
Notes: The sslKey file must be created manually in order for SSL to work.
Example: sslKeyFile = "/IN/service_packages/OSD/sslKey.pem"

tssfTimeOutSecs
Syntax: tssfTimeOutSecs = seconds
Description: The maximum period osdInterface should wait for a response from slee_acs before giving up and sending a SOAP fault.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: 
Default: 10
Notes: 
Example: tssfTimeOutSecs = 20

validateRequestNameSpace
Syntax: validateRequestNameSpace = true|false
Description: Do not allow incorrect WSDL namespace.
Type: Boolean
Optionality: Optional (default used if not set).
Allowed: 
- true - namespace in the request tag of the incoming request will be ignored.
false - the namespace must match the name space of the operation set.

Default: false
Notes: validateRequestNameSpace = true

databaseCachingRules

Syntax: databaseCachingRules = {rules}
Description: The list of table re-read time rules.
Type: List
Optionality: Optional (default used if not set).
Allowed: operationSetsDataExpirySecs, operationDataExpirySecs, clientAspDataExpirySecs, portListsDataExpirySecs, acsProfileDataExpirySecs
Default: operationSetsDataExpirySecs = 10, operationDataExpirySecs = 10, clientAspDataExpirySecs = 10, portListsDataExpirySecs = 10, acsProfileDataExpirySecs = 10
Notes: databaseCachingRules = {
  operationSetsDataExpirySecs = 10
  operationDataExpirySecs = 10
  clientAspDataExpirySecs = 10
  portListsDataExpirySecs = 10
  acsProfileDataExpirySecs = 10
}

Syntax: acsProfileDataExpirySecs = seconds
Description: The maximum age data from ACS_PROFILE_DETAILS table can be before it is re-read from the database.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: 10
Default: 10
Notes: Member of the databaseCachingRules (on page 33) section.
Example: acsProfileDataExpirySecs = 300

Syntax: clientAspDataExpirySecs = seconds
Description: The maximum age data from OSD_CLI_ASP and OSD_CLIENT_ASP_ACCESS tables can be before it is re-read from the database.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: 10
Default: 10
Notes: Member of the databaseCachingRules (on page 33) section.
Example: clientAspDataExpirySecs = 300

operationDataExpirySecs

Syntax: operationDataExpirySecs = seconds
Description: The maximum age data from OSD_OPERATION table can be before it is re-read from the database.
Type: Integer
Optionality: Optional (default used if not set).
Chapter 4

**Allowed:**

**Default:** 10

**Notes:** Member of the databaseCachingRules (on page 33) section.

**Example:** operationDataExpirySecs = 300

---

**operationSetsDataExpirySecs**

**Syntax:** operationSetsDataExpirySecs = seconds

**Description:** The maximum age data from OSD_OPERATION_SET table can be before it is re-read from the database.

**Type:** Integer

**Optionality:** Optional (default used if not set).

**Allowed:**

**Default:** 10

**Notes:** Member of the databaseCachingRules (on page 33) section.

**Example:** operationSetsDataExpirySecs = 300

---

**portListsDataExpirySecs**

**Syntax:** portListsDataExpirySecs = seconds

**Description:** The maximum age data from OSD_PORT_LIST and OSD_PORT_LIST_ENTRY tables can be before it is re-read from the database.

**Type:** Integer

**Optionality:** Optional (default used if not set).

**Allowed:**

**Default:** 10

**Notes:** Member of the databaseCachingRules (on page 33) section.

**Example:** portListsDataExpirySecs = 300

---

**NumberRules**

**Syntax:** NumberRules = [{rule_1},{rule_2}...]

**Description:** Rules for denormalizing numbers to send to slee_acs in an IDP.

**Type:** Array of number rules.

**Optionality:** Optional (default used if not set).

**Allowed:**

**Default:**

```json
NumberRules = [
  { prefix="", min=0, max = 100, remove=0, prepend="", resultNoa=4 }]
```

**Notes:** The rules below assume that numbers in the XML start with a country code and should be sent in international format (NOA= 4).

**Example:**

```json
NumberRules = [
  { prefix="", min=0, max = 100, remove=0, prepend="", resultNoa=4 }]
```

---

**max**

**Syntax:** max = len

**Description:** The maximum number length.

**Type:** Integer

**Optionality:** Optional (default used if not set).

**Allowed:**
Configuration

Default: 999
Notes: Used in number normalization and denormalization (on page 34) rules.
Example: max = 32

min
Syntax: min = len
Description: The minimum number length.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: Default: 0
Notes: Used in number normalization and denormalization (on page 34) rules.
Example: min = 4

prefix
Syntax: prefix = "digit"
Description: This rule is applied to numbers with this prefix.
Type: String
Optionality: Optional
Allowed: One or more decimal digits
Default: Notes: Used in number normalization and denormalization (on page 34) rules.
Example: prefix = "25"

prepend
Syntax: prepend= "digits"
Description: Determines the digits that are to be prepended to the number, after stripping any as specified previously.
Type: String
Optionality: Allowed: Default: Notes: Used in number normalization and denormalization (on page 34) rules.
Example: prepend = "1111"

remove
Syntax: remove = num
Description: Determines the number of digits that are stripped from the beginning of the number.
Type: Integer
Optionality: Allowed: Default: Notes: Used in number normalization and denormalization (on page 34) rules.
Example: remove = 2
resultNoa
Syntax: \[\text{resultNoa} = \text{noa}\]
Description: Resulting NOA after the normalization.
Type: 
Optionality: 
Allowed: 
Default: 
Notes: Used in number normalization and \textit{denormalization} (on page 34) rules.
Example: \[\text{resultNoa} = 4\]

OSD SMS parameters
Here are the parameters in the OSD section of the eserv.config.
Note: The following parameters are only relevant on an SMS.
OSD = {
  wsd1UriBaseName = "http://nzwn-test08.uk.oracle.com/wsdls"
  useDeprecatedSchema = false
  maxProfileDetailsAge = 60
  maxServiceHandleAge = 60
  osdWsdlRegenerator = {
    waitTimeSecs = 2
    useOracleAlerts = false
    oracledatabase = "nzwn-test08_SMF"
  }
}

maxProfileDetailsAge
Syntax: \[\text{maxProfileDetailsAge} = \text{seconds}\]
Description: The maximum age data from ACS_PROFILE_DETAILS table can be before it is re-read from the database.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: 
Default: 60
Notes: 
Example: \[\text{maxProfileDetailsAge} = 80\]

maxServiceHandleAge
Syntax: \[\text{maxServiceHandleAge} = \text{seconds}\]
Description: The maximum age data from OSD_SERVICE_HANDLE and OSD_MANDATORY_INPUT_PARAMETER tables can be before data is re-read from the database.
Type: Integer
Optionality: Optional (default used if not set).
Allowed: 
Default: 60
Notes: 

36  NCC Open Services Development User's and Technical Guide
maxServiceHandleAge = 80

oracledatabase

Syntax: \texttt{oracledatabase} = "database\_name"

Description: The remote Oracle database to connect to, from tnsnames.ora.

Type: String

Optionality: Optional (default used if not set).

Allowed: None

Notes: Usually not specified as it is usual to use the local database.

Example: oracledatabase = "nzwn-test08_SMF"

osdWsdlRegenerator

Syntax: \texttt{osdWsdlRegenerator} = \{parameter\_list\}

Description: The osdWsdlRegenerator section is a list of parameters that govern when a WSDL file that has changed is compiled.

Type: List

Optionality: Mandatory

Allowed: None

Notes: osdWsdlRegenerator = \{
  waitForTimeSecs = 2
  useOracleAlerts = false
\}

useDeprecatedSchema

Syntax: \texttt{useDeprecatedSchema} = \texttt{true|false}

Description: Set true to force use of deprecated old product namespace and schema.

Type: Boolean

Optionality: Optional (default used if not set).

Allowed: true, false

Default: false

Notes: False means using the OCNCC namespace and schema.

Example: useDeprecatedSchema = false

useOracleAlerts

Syntax: \texttt{useOracleAlerts} = \texttt{true|false}

Description: Whether to use Oracle alerts.

Type: Boolean

Optionality: Mandatory

Allowed: true - waiting for an Oracle alert

  • false - sleep and re-read OSD\_CHANGED\_OPERATION\_SET

Default: None

Notes: None

Example: useOracleAlerts = false
waitTimeSecs

Syntax: waitTimeSecs = seconds
Description: The number of seconds to wait for an Oracle alert or to sleep, depending on the useOracleAlerts setting.
Type: Integer
Optionality: Mandatory
Allowed: None
Default: None
Notes: Example: waitTimeSecs = 2

wsdlUriBaseName

Syntax: wsdlUriBaseName = "location"
Description: The URL of the directory containing the OCNCC.xsd file.
Type: String
Optionality: Required
Allowed: Default: http://IP_of_SMS/wsdls
Notes: This must match the corresponding value for the WSDLURL parameter in the sms.jnlp file.
Example: wsdlUriBaseName = "http://nzwn-test08.uk.oracle.com/wsdls"

High volume configuration

If a large number of XML parameters are expected to be passed between osdInterface and slee_acs
then the following configuration should be done:
1. Add a number of large SLEE events to SLEE.cfg (to get 500 20K events)
   MAXEVENTS=500 20480
2. Add the following to the acsChassis section of acs.conf
   minimumSizeOfConnectSleeEvent 20480

acs.conf configuration

osd acs.conf configuration

The control plan compiler requires the plug-in libwsdlGenerator.so for generating the WSDL file and attaching the control plan to the OSD operation. This configuration is inserted in to the acs.conf file on the SMS during installation.

An example acsCompilerDaemon section is shown below.

acsCompilerDaemon
  CompilerPlugin libwsdlGenerator.so
  alertTimeout 3
  maxBranches 99
  maxNodes 2000
  maxCompiledKb 256
  compressAtKb 128
  compressLevel 1:
See ACS Technical Guide for more acs.conf information.

**SLEE.config Configuration**

**osd SLEE.config configuration**

When OSD is installed, the SLEE.cfg file has the following line added by osdScp:

```bash
INTERFACE=osdInterface osdInterface.sh /IN/service_packages/OSD/bin EVENT
```

However, if you wanted to loadshare across several osdInterface processes on the same SLC, then a line per interface is required, for example, loadsharing across three interfaces would require:

```bash
INTERFACE=osdIf1 osdInterface1.sh /IN/service_packages/OSD/bin EVENT
INTERFACE=osdIf2 osdInterface2.sh /IN/service_packages/OSD/bin EVENT
INTERFACE=osdIf3 osdInterface3.sh /IN/service_packages/OSD/bin EVENT
```

**Note:** The interface names in the SLC ports table would be then be osdIf1, osdIf2 and osdIf3.

**sms.jnlp Configuration**

**WSDL parameters**

The two applet parameters that are required for WSDL file generation, are listed below. These parameters are automatically added to the sms.jnlp file when you install OSD:

- **WSDLDirectory**
- **WSDLURL**

For more information about applet parameters in .jnlp files, see Customizing the screens in SMS Technical Guide.

**WSDLDirectory**

**Syntax:**

```
<param name="WSDLDirectory" VALUE="/IN/html/wsdls" />
```

**Description:**

This is set on install to this value which forms the first part of the Operation Sets WSDL Location field value.

**Type:** String

**Optionality:** Required

**Allowed:**

**Default:**

**Notes:**

Part of OSD.

If this parameter value is changed, the parameter `wsdlUriBaseName` in the eserv.config file must also be changed.

**Example:**

```
<param name="WSDLDirectory" VALUE="/IN/html/wsdls" />
```

**WSDLURL**

**Syntax:**

```
<param name="WSDLURL" value="url" />
```

**Description:**

This is set to the WSDL URL field value (same as `wsdlUriBaseName` parameter), and has the form of:

```
http://host_name/wsdls
```

**Type:** String

**Optionality:** Optional (default used if not set).

**Allowed:**
Default: Part of OSD.

Notes: If this parameter value is changed, the parameter `wsdlUriBaseName` in the `eserv.config` file must also be changed.

Example: `<param name="WSDLURL" VALUE="http://nzwn-test08.uk.oracle.com/wsdls" />`
Overview

Introduction

This chapter explains the processes which run automatically as part of the application. These processes are started automatically by one of the following:

- inittab
- crontab
- Service Logic Execution Environment SLEE

Note: This chapter also includes some plug-ins to background processes which do not run independently.

In this chapter

This chapter contains the following topics.

- osdInterface 41
- WSDL Generating Plug-in 43
- WSDL Regenerator 44
- Statistics Logged 44
- Reports 44

osdInterface

About the osdInterface

The Open Services Development (OSD) osdInterface is a SLEE interface that accepts SOAP requests from ASPs across a configurable range of TCP/IP ports, and transforms them into IDPs. The IDPs are sent to slee_acs in SLEE events, and SOAP responses are then sent to the requesting clients based on the result of the control plan executed in response to the IDP.

The osdInterface also supports heartbeat messages, received in the form of PING requests, and sends a suitable response before closing the connection. The PING messages are used to check if the server is on-line and therefore able to process requests.

The osdInterface rejects requests that do not contain the mandatory parameters, listed in the table below, for the service being used. The interface returns a fault with errorCode 3 ("Missing parameter") when it rejects a request.

Mandatory Parameters for OSD

This table lists the services that require mandatory parameters, and the mandatory parameters for each service.

<table>
<thead>
<tr>
<th>Service</th>
<th>Mandatory Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACS</td>
<td>CC Service Number</td>
</tr>
<tr>
<td>ACS_Management</td>
<td>CC Service Number</td>
</tr>
</tbody>
</table>
### Caches

The osdInterface maintains a cache of configuration objects largely sourced from the database. Some caches will be augmented with runtime information NOT sourced from the database, but needed to be stored on a per-object basis. Caching is carried out according to the `databaseCachingRules` configuration rules. See `databaseCachingRule` (on page 33)s.

This table lists the caches.

<table>
<thead>
<tr>
<th>Cache</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASP</td>
<td>Generally sourced from OSD_CLIENT_ASP and OSD_CLIENT_ASP_ACCESS</td>
</tr>
<tr>
<td>OperationSet</td>
<td>Generally sourced from OSD_OPERATION_SET</td>
</tr>
<tr>
<td>Operation</td>
<td>Fully sourced from OSD_OPERATION</td>
</tr>
<tr>
<td>ProfileTag</td>
<td>Fully sourced from ACS_PROFILE_DETAILS and ACS_TAG_TO_PROFILE_MAPPING</td>
</tr>
<tr>
<td>SOAPPort</td>
<td>This will be sourced from OSD_PORT_LIST and OSD_PORT_LIST_ENTRY</td>
</tr>
</tbody>
</table>

### Throttling

Throttling will be achieved by allocating a series of counters, each one representing a second.

---

NCC Open Services Development User's and Technical Guide
As new transactions are executed the throttle mechanism increments the counter in the slot of the current (most recent) second. As time rolls over it discards slots older than the configurable measurement period \( \text{rateCalculationPeriodSecs} \) (on page 31).

To check that traffic is within tolerances the throttle mechanism simply refers to the sum of the counters.

**SOAP Requests**

The osdInterface will bind to the address/port of all connections defined, and process SOAP requests on all those that are not quiescing (that is, all those that have not been dropped from the configuration table since osdInterface last loaded them from there).

When a connection is dropped from the configuration it will not be removed from the cache at the next cache expiry, but merely updated so that its isQuiescing variable is TRUE. A quiescing connection will be destroyed (and removed from the cache) only when its number of outstanding requests falls to zero.

The SOAP document is parsed and extracted tags of relevance are passed to osdInterface.

**Note:** Not all tags encountered in the document need be understood, or used.

**Restrictions**

The following restrictions apply to this version of the interface:

- osdInterface receives SOAP requests over HTTP 1.1 over HTTP or HTTPS
- A single SOAP request per connection will be accepted
- Although the WSDL files specify xs:dateTime for some date types, timezones other than GMT are not supported by osdInterface, that is, the only valid date time format is: \( CCYY-MM-DDThh:mm:ssZ \)

**Heartbeat message**

The OSD interface supports ping requests occasionally sent by third-party systems such as M-POS, and responds appropriately before closing the connection.

An HTTP 1.1 POST message with the message body \(<\text{PING}/\rangle\) is interpreted as a heartbeat message. This message will be responded to with an HTTP 1.1 code 204 response (no content) before closing the connection.

**Example ping request:**

```
POST http://hostname/ HTTP/1.1
... header fields ...
<PING/>
```

**Example OSD response:**

```
HTTP/1.1 204 No Content
```

**WSDL Generating Plug-in**

**Overview**

This plug-in is used to generate WSDL files from control plan compilation. The plug-in must be configured in `acs.conf` as:

```
CompilerPlugin libwsdlGenerator.so
```

The compiler daemon executes each plug-in in the order they appear in the configuration file, so the order may become important when more than this plug-in is available.
Note: libwsdlGenerator.so is also used for OSD screen configuration changes to regenerate the WSDL file.

WSDL Regenerator

Overview
The WSDL regenerator is used to re-compile WSDL files after any changes made to port lists, operation sets or operations through the OSD screens.
The re-compile is performed using libwsdlGenerator.so.

Statistics Logged

Overview
Statistics are collected using the smsStats statistics methods at appropriate points in the code.
These sets of statistics are gathered to provide the information necessary for the reports.

OSD statistics
This table lists the statistics collected.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASP_REQUESTS_PROCESSED_SUCCESSFULLY</td>
<td>The number of successful SOAP requests from each Client ASP.</td>
</tr>
<tr>
<td>ASP_REQUESTS_RECEIVED</td>
<td>The number of SOAP requests from each Client ASP.</td>
</tr>
<tr>
<td>ASP_UNSUCCESSFUL_REQUESTS</td>
<td>The number of unsuccessful SOAP requests from each Client ASP.</td>
</tr>
<tr>
<td>TOTAL_REQUESTS_PROCESSED_SUCCESSFULLY</td>
<td>The number of successful SOAP requests, for the system.</td>
</tr>
<tr>
<td>TOTAL_REQUESTS_RECEIVED</td>
<td>The number of SOAP requests, for the system.</td>
</tr>
<tr>
<td>TOTAL_UNSUCCESSFUL_REQUESTS</td>
<td>The number of unsuccessful SOAP requests, for the system.</td>
</tr>
</tbody>
</table>

Reports

Overview
The collected statistics are used for reports:
- osd Asp
- osd System
This topic covers what these reports are. For information on generating these reports see the Report Functions topic in SMS User's Guide.

Report Columns
The columns for the reports show:
Chapter 5

- The reporting interval
- The peak requests for either Client ASP or system.
- The total number of requests for either Client ASP or system.
- The total number of failed requests for either Client ASP or system.

Peak requests are defined as the maximum number of requests in any on stats collection period.

For example, if the reporting interval is one hour and the stats reporting period is five minutes then the peak requests is the maximum number of requests in any one five minute period during the hour.

**osd ASP report**

This report provides details on SOAP requests by client ASP.

**Report parameters**
The table below describes the SMS parameters for this report.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASP</td>
<td>The client ASP name, min of 1, max of 64 characters.</td>
</tr>
<tr>
<td>Reporting Period</td>
<td>Day, week, month reporting.</td>
</tr>
<tr>
<td>Interval</td>
<td>Granularity of report lines, min 10 minute intervals.</td>
</tr>
<tr>
<td>Start Date</td>
<td>Start date of report. Omitted reports on all statistics collected.</td>
</tr>
<tr>
<td>Hours Since</td>
<td>End date of report. Omitted reports on all statistics collected after Start Date.</td>
</tr>
</tbody>
</table>

**Note:** When dates are used, the format is:

```
YYYYMMDD[HH24][MM][SS]
```

Examples:

- 20090823
- 2009092312
- 200908231225
- 20090823122533
Report example
Here is an example of the OSD ASP Report.

```
OSD ASP Report
------------------------
Start Date: 01 January 2008
Finish Date: 01 January 2009
Report Type: Year
ASP: boss

17 August 2009, 21:13:23
```

<table>
<thead>
<tr>
<th>Interval</th>
<th>Peak Requests</th>
<th>Total Requests</th>
<th>Total Failed Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td>20080101000000</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101000200</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101010000</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101010040</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101010050</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101011000</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101011010</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101011020</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101011030</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101011050</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101012000</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20080101012020</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**osd System report**
This report provides details on SOAP requests for the system.

Report parameters
The table below describes the SMS parameters for this report.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reporting Period</td>
<td>Day, week, month reporting.</td>
</tr>
<tr>
<td>Interval</td>
<td>Granularity of report lines, min 10 minute intervals.</td>
</tr>
<tr>
<td>Start Date</td>
<td>Start date of report. Omitted reports on all statistics collected.</td>
</tr>
<tr>
<td>Hours Since</td>
<td>End date of report. Omitted reports on all statistics collected after Start Date.</td>
</tr>
</tbody>
</table>

**Note:** When dates are used, the format is:
```
YYYYMMDD[HH24][M][SS]]
```

Examples:
- 20090823
- 2009092312
- 200908231225
- 20090823122533
### Report Example

**OSD System Report**

---

**Start Date:** 01 July 2009  
**Finish Date:** 01 August 2009  
**Report Type:** Month

17 August 2009, 21:30:26

<table>
<thead>
<tr>
<th>Interval</th>
<th>Peak Requests</th>
<th>Total Requests</th>
<th>Total Failed Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td>200907010000</td>
<td>□</td>
<td>□</td>
<td>0</td>
</tr>
<tr>
<td>200907010200</td>
<td>□</td>
<td>□</td>
<td>0</td>
</tr>
<tr>
<td>200907010400</td>
<td>□</td>
<td>□</td>
<td>0</td>
</tr>
<tr>
<td>200907010600</td>
<td>□</td>
<td>□</td>
<td>0</td>
</tr>
<tr>
<td>200907010800</td>
<td>□</td>
<td>□</td>
<td>0</td>
</tr>
<tr>
<td>200907011000</td>
<td>□</td>
<td>□</td>
<td>0</td>
</tr>
</tbody>
</table>
Overview

Introduction

This chapter explains the important processes on each of the server components in the NCC, and a number of example troubleshooting methods which will help aid the troubleshooting process before raising a support ticket.

In this chapter

This chapter contains the following topics.

Common Troubleshooting Procedures

Common Troubleshooting Procedures

Introduction

Refer to NCC System Administrator's Guide for troubleshooting procedures common to all NCC components.

Checking current processes

You can check which processes are running using the standard UNIX command: ps. To find processes being run by Oracle software, you can grep for the string ‘oper’, which will display all processes being run by the application operator accounts (for example, acsOper, ccsOper and smfOper).

Note: Some processes which are required for proper functioning may be run by other users, including root or the user which runs the webserver.

Example command: ps -ef | grep oper

For more information about the ps command, see the system documentation for the ps command.

You can also check how much of the processor a process is using by running the standard UNIX tool: top. If you have some baseline measurements, you will be able to compare it with the current load.

Example command: top

Tip: Some processes should only have one instance. If there are two or more instances, this may indicate a problem. For example, there will usually only be one timerIF running on each SLC.

For more information about which processes should be running on each node, check the Process List for each node in Installation.
HTTP Error Codes

The HTTP Error Codes are listed here.

<table>
<thead>
<tr>
<th>Code</th>
<th>HTTP Code Meaning</th>
<th>Meaning</th>
<th>Recovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>HTTP_CODE_OK</td>
<td>Normal response.</td>
<td>None Required.</td>
</tr>
<tr>
<td>204</td>
<td>HTTP_CODE_NO_CONTENT</td>
<td>The osdInterface received a &lt;PING/&gt; request. If &lt;PING/&gt; appears in the body of the request, the osdInterface will respond with a 204.</td>
<td>Remove &lt;PING/&gt; from the request, and retry.</td>
</tr>
<tr>
<td>400</td>
<td>HTTP_CODE_BAD_REQUEST</td>
<td>The request could not be parsed. Either the HTTP headers are invalid, the data does not fit inside internal buffers, or the body does not start with &lt;?xml.</td>
<td>If the client is sending a valid SOAP request, this indicates a fault. Otherwise, correct the request syntax and retry the request.</td>
</tr>
<tr>
<td>401</td>
<td>HTTP_CODE_AUTHORIZATION_REQUIRED</td>
<td>OSD requires ASPs to authenticate using Basic Access Authentication.</td>
<td>The client should retry the request with a valid Authorization header.</td>
</tr>
<tr>
<td>404</td>
<td>HTTP_CODE_NOT_FOUND</td>
<td>The request namespace was not found, and the request could not be processed.</td>
<td>Correct the namespace specification. It currently must be placed on the SOAP Operation Request element.</td>
</tr>
<tr>
<td>500</td>
<td>HTTP_CODE_INTERNAL_SERVER_ERROR</td>
<td>The server encountered an internal problem and could not continue to process the request.</td>
<td>Usually indicates an internal software fault. Reconciliation may be required.</td>
</tr>
<tr>
<td>503</td>
<td>HTTP_CODE_UNAVAILABLE</td>
<td>The service is unable to process the ASPs request at this time. The body of the message may offer additional information.</td>
<td>Correct the problem indicated in the response body and retry.</td>
</tr>
</tbody>
</table>

SOAP release causes

The SOAP release causes are listed here.

<table>
<thead>
<tr>
<th>Cause</th>
<th>OSD Meaning</th>
<th>Meanings</th>
<th>Recovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No such subscriber</td>
<td>User specified through disconnect node in control plan</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>No such service</td>
<td>User specified through disconnect node in control plan</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Missing parameter</td>
<td>User specified through disconnect node in control plan</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Mis-typed parameter</td>
<td>User specified through disconnect node in control plan</td>
<td></td>
</tr>
<tr>
<td>Cause</td>
<td>OSD Meaning</td>
<td>Meanings</td>
<td>Recovery</td>
</tr>
<tr>
<td>-------</td>
<td>---------------------------------</td>
<td>--------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5</td>
<td>System error</td>
<td>User or software specified. The system encountered an unexpected issue processing the response. This is a catch all error, and generally indicates a software fault.</td>
<td>The request may or may not have been completed, reconciliation may be required once the software fault is fixed. The request may or may not work if retried.</td>
</tr>
<tr>
<td>6</td>
<td>Operation does not exist</td>
<td>Software specified. The operation specified in the inbound OSD request is not configured on the SLC.</td>
<td>Configure the operation on the SMS.</td>
</tr>
<tr>
<td>7</td>
<td>Operation not available</td>
<td>Software specified. The operation specified in the inbound OSD request is not enabled.</td>
<td>Enable the operation on the SMS and retry the request.</td>
</tr>
<tr>
<td>8</td>
<td>Invalid transaction type</td>
<td>Software specified. UNUSED</td>
<td>UNUSED</td>
</tr>
<tr>
<td>9</td>
<td>Transaction not found</td>
<td>Software specified. UNUSED</td>
<td>UNUSED</td>
</tr>
<tr>
<td>10</td>
<td>No response from ACS</td>
<td>Software specified – transient error only, retry permitted. At the time the response was sent, ACS had not provided a response to the interface. This will happen when the system is slow or when it is being shut down.</td>
<td>If it is valid to execute the request multiple times, retry the request. Otherwise, use an agreed reconciliation process.</td>
</tr>
<tr>
<td>11</td>
<td>Too many transactions</td>
<td>Software specified – transient error only, retry permitted. UNUSED</td>
<td>UNUSED</td>
</tr>
<tr>
<td>12</td>
<td>Duplicate transaction</td>
<td>Software specified. UNUSED</td>
<td>UNUSED</td>
</tr>
</tbody>
</table>
| 13    | Cannot Parse SOAP envelope      | Software specified. The request was invalid because:  
  1  The XML was not valid XML  
  2  The service key for the operation could not be found | Ensure that the XML being presented to the service is valid, and that the SLC configuration is correct. Once this is confirmed, retry the request. |
| 14    | Operation disabled              | Software specified. The ASP has insufficient permissions to execute that operation. | Grant access to the operation using the SMS screens and retry the request. |

**Note:** Unless otherwise specified, these error causes are permanent failures and retry should not be attempted by the ASP.

These causes are used in the errorCode parameter of SOAP faults sent to ASPs when failures occur.

For example, if there is a profile branching node on something from Incoming Session data and the data not found branch is used, then there can be a Disconnect Node with cause 3. Then, OSD interface will send a SOAP fault with error code 3 and the ASP will know that the SOAP request has been rejected because of a missing parameter.
Chapter 7

About Installation and Removal

Overview

Introduction

This chapter provides details of the installation and removal process for the application.

In this chapter

This chapter contains the following topics.

Installation and Removal Overview 53
Post Install Replication 54

Installation and Removal Overview

Introduction

For information about the following requirements and tasks, see *NCC Installation Guide*:

- NCC system requirements
- Pre-installation tasks
- Installing and removing NCC packages

OSD packages

An installation of OSD includes the following packages, on the:

- SMS:
  - osdSms
  - osdCluster (for clustered SMS)
- SLC:
  - osdScp

Known issue

Once installation is complete on both nodes of a cluster, the following error may appear:

WARNING: On node <host name> resource group OsdWsd1Regenerator-harg is online but the monitor of resource OsdWsd1Regenerator-hars failed to start

If this happens, enter the following command line to fix the problem:

```bash
scswitch -z -g OsdWsd1Regenerator-harg -h The_Other_Host_Name
```
Post Install Replication

Additional replication

As part of the OSD installation the ACS_TAG_TO_PROFILE_MAPPING table replication is added to:
Operator Functions > Node Management > Table Replication > Apps > Acs_Service.

After installation, please ensure ACS_TAG_TO_PROFILE_MAPPING and ACS_PROFILE_DETAIL are replicated to all SLC nodes.

OSD Additional replication

Follow these steps to configure SMS replication to ensure all OSD definitions are available on all SLCs nodes which are running OSD.

For more information about how to complete these steps, see SMS User's Guide.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Restart SMS Java (Swing) Administration screens.</td>
</tr>
<tr>
<td>2</td>
<td>Open the Table Replication tab on the SMS Node Management screen.</td>
</tr>
<tr>
<td>3</td>
<td>Add the following OSD tables to the SLC nodes which have OSD installed:</td>
</tr>
<tr>
<td></td>
<td>• OSD_CLIENT_ASP</td>
</tr>
<tr>
<td></td>
<td>• OSD_CLIENT_ASP_ACCESS</td>
</tr>
<tr>
<td></td>
<td>• OSD_MANDATORY_INPUT_PARAMETER</td>
</tr>
<tr>
<td></td>
<td>• OSD_OPERATION</td>
</tr>
<tr>
<td></td>
<td>• OSD_OPERATION_SET</td>
</tr>
<tr>
<td></td>
<td>• OSD_PORT_LIST</td>
</tr>
<tr>
<td></td>
<td>• OSD_PORT_LIST_ENTRY</td>
</tr>
<tr>
<td></td>
<td>• OSD_SERVICE_HANDLE</td>
</tr>
<tr>
<td>4</td>
<td>Save the updated node config by clicking Save.</td>
</tr>
<tr>
<td>5</td>
<td>Click Create Config File update the SLCs.</td>
</tr>
</tbody>
</table>
NCC Glossary of Terms

AAA

ACS
Advanced Control Services configuration platform.

ANI
Automatic Number Identification - Term used in the USA by long-distance carriers for CLI.

ASP
- Application Service Provider, or

CC
Country Code. Prefix identifying the country for a numeric international address.

CCS
1) Charging Control Services (or Prepaid Charging) component.
2) Common Channel Signalling. A signalling system used in telephone networks that separates signalling information from user data.

CLI
Calling Line Identification - the telephone number of the caller. Also referred to as ANI.

CPE
Control Plan Editor (previously Call Plan Editor) - software used to define the logic and data associated with a call -for example, "if the subscriber calls 0800 nnnnnn from a phone at location xxx then put the call through to bb bbb bbbb".

cron
Unix utility for scheduling tasks.

crontab
File used by cron.

Diameter
A feature rich AAA protocol. Utilises SCTP and TCP transports.

DP
Detection Point
DTMF

Dual Tone Multi-Frequency - system used by touch tone telephones where one high and one low frequency, or tone, is assigned to each touch tone button on the phone.

HTML

HyperText Markup Language, a small application of SGML used on the World Wide Web.
It defines a very simple class of report-style documents, with section headings, paragraphs, lists, tables, and illustrations, with a few informational and presentational items, and some hypertext and multimedia.

HTTP

Hypertext Transport Protocol is the standard protocol for the carriage of data around the Internet.

IDP

INAP message: Initial DP (Initial Detection Point)

IN

Intelligent Network

INAP

Intelligent Network Application Part - a protocol offering real time communication between IN elements.

Initial DP

Initial Detection Point - INAP Operation. This is the operation that is sent when the switch reaches a trigger detection point.

IP

1) Internet Protocol
2) Intelligent Peripheral - This is a node in an Intelligent Network containing a Specialized Resource Function (SRF).

IP address

Internet Protocol Address - network address of a card on a computer

Messaging Manager

The Messaging Manager service and the Short Message Service components of Oracle Communications Network Charging and Control product. Component acronym is MM (formerly MMX).

MM

Messaging Manager. Formerly MMX, see also XMS (on page 58) and Messaging Manager (on page 56).

MO

Mobile Originated
MT
Mobile Terminated

NOA
Nature Of Address - a classification to determine in what realm (Local, National or International) a given phone number resides, for the purposes of routing and billing.

Oracle
Oracle Corporation

SCTP
Stream Control Transmission Protocol. A transport-layer protocol analogous to the TCP or User Datagram Protocol (UDP). SCTP provides some similar services as TCP (reliable, in-sequence transport of messages with congestion control) but adds high availability.

Service Provider
See Telco.

Session
Diameter exchange relating to a particular user or subscriber access to a provided service (for example, a telephone call).

SGML

SLC
Service Logic Controller (formerly UAS).

SLEE
Service Logic Execution Environment

SMS
Depending on context, can be:
- Short Message Service
- Service Management System platform
- NCC Service Management System application

SOAP
SRF

Specialized Resource Function - This is a node on an IN which can connect to both the SSP and the SLC and delivers additional special resources into the call, mostly related to voice data, for example play voice announcements or collect DTMF tones from the user. Can be present on an SSP or an Intelligent Peripheral (IP).

SSP

Service Switching Point

System Administrator

The person(s) responsible for the overall set-up and maintenance of the IN.

TCP

Transmission Control Protocol. This is a reliable octet streaming protocol used by the majority of applications on the Internet. It provides a connection-oriented, full-duplex, point to point service between hosts.

Telco

Telecommunications Provider. This is the company that provides the telephone service to customers.

Telecommunications Provider

See Telco.

URL

Uniform Resource Locator. A standard way of specifying the location of an object, typically a web page, on the Internet.

VWS

Oracle Voucher and Wallet Server (formerly UBE).

WSDL

Web Services Description Language.

XML

eXtensible Markup Language. It is designed to improve the functionality of the Web by providing more flexible and adaptable information identification.

It is called extensible because it is not a fixed format like HTML. XML is a `metalanguage’ — a language for describing other languages—which lets you design your own customized markup languages for limitless different types of documents. XML can do this because it's written in SGML.

XMS

Three letter code used to designate some components and path locations used by the Oracle Communications Network Charging and Control Messaging Manager (on page 56) service and the Short Message Service. The published code is MM (on page 56) (formerly MMX).
Index

A
AAA • 55
About Installation and Removal • 53
About the osdInterface • 41
About This Document • v
ACS • 55
acs.conf configuration • 39
Add Allowed Operations • 24
Adding SLC ports • 18
Additional replication • 54
ANI • 55
applicationContext • 29
Architecture • 2
ASP • 55
Audience • v

B
Background Processes • 41
basicRealm • 29

C
Cached objects • 42
CC • 55
CCS • 55
CCS service handlers • 11
Checking current processes • 49
CLI • 55
Client ASPs • 23
Client ASPs fields • 24
Client ASPs tab • 5, 23
Common Troubleshooting Procedures • 49
Configuration • 27
Configuration file format • 27
connectionTimeOutSecs • 30
Control Plan • 7
Control plan compilation • 8
Copy Node Configuration • 8
Copyright • ii
CPE • 55
cron • 55
crontab • 55

D
databaseCachingRules • 33, 34, 42
Diameter • 55
Document Conventions • vi
DP • 55
DTMF • 56

E
Edit Client ASPs • 24
Edit Service Providers • 18
Editing Operation Sets • 21
Editing Operations • 23
Editing SLC ports • 18, 19
Editing the file • 28
eserv.config Configuration • 27
eserv.config files delivered • 28
Example Find screen • 16
Example scenario • 3

F
Find Screens • 15, 21, 23, 24

G
Getting Started • 13

H
Heartbeat message • 43
High volume configuration • 39
HTML • 56
HTTP • 56
HTTP Error Codes • 50

I
IDP • 56
IN • 56
INAP • 56
Incoming SOAP message • 3, 11
Initial DP • 56
Installation and Removal Overview • 53
Introduction • 1, 2, 11, 15, 16, 19, 27, 49, 53
IP • 56
IP address • 56

K
Known issue • 53

L
Loading eserv.config changes • 28
loadReportingPeriod • 30

M
Mandatory Parameters for OSD • 41
max • 35
maxProfileDetailsAge • 37
maxServiceHandleAge • 37
Messaging Manager • 56, 58
min • 35
MM • 56, 58
MO • 56
MT • 57

N
NOA • 57

O
Operation Set WSDL • 9
XMS • 56, 58