THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCBs public domain version of the UNIX operating system. All rights reserved. Copyright 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: http://www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Copyright 2015 Cisco Systems, Inc. All rights reserved.
# Contents

Contents ......................................................................................................................................................... i  

Preface ............................................................................................................................................................ xii  
  Purpose ....................................................................................................................................................... xii  
  Audience .................................................................................................................................................... xii  
  Organization ................................................................................................................................................ xii  
  Terms and Definitions ................................................................................................................................. xiii  
  Product Naming Conventions .................................................................................................................. xiv  
  Conventions ............................................................................................................................................... xv  
  Obtaining Documentation and Submitting a Service Request ............................................................... xvi  
  Field Alerts and Field Notices ................................................................................................................... xvi  
  Documentation Feedback ........................................................................................................................... xvi  

1. Introduction to Web Services .................................................................................................................. 1  
  Unified CCMP Overview ............................................................................................................................. 1  
  Web Service Protocols ............................................................................................................................... 1  
    SOAP ....................................................................................................................................................... 2  
    REST ....................................................................................................................................................... 2  
  Security ..................................................................................................................................................... 3  
    Resource Management and Subscriptions Web Services .................................................................. 3  
      SOAP .................................................................................................................................................... 4  
      REST .................................................................................................................................................... 5  
    Notifications ......................................................................................................................................... 5  
    More Information .................................................................................................................................. 5  

2. Common Data Types ............................................................................................................................... 6  
  Remote Resource Types ............................................................................................................................ 6  
    Provisionable Remote Resource Types ................................................................................................. 6
### Contents

Non Provisionable Remote Resource Types ........................................ 7

**System Resource Types** .................................................................. 10

- Provisionable System Resource Types ........................................ 10
- Non Provisionable System Resource Types ................................. 10

**SCD Effective Dating** ................................................................. 12


**Overview** .................................................................................. 13

- Description ............................................................................. 13
- Concepts ............................................................................... 13
- Item Types ............................................................................ 14
- Pkey Map Item Types ........................................................... 14
- Methods ............................................................................... 15
- About Enterprise-Level Caching ............................................... 16

**Specification** ............................................................................. 16

- SOAP .................................................................................. 16
- REST ................................................................................... 17

**Restrictions** ............................................................................... 17

- Array Limits .......................................................................... 17
- REST URI Segment Limit ....................................................... 17

**Errors** .................................................................................. 18

**Data Types** ............................................................................... 21

- Resource Hierarchy Description ........................................... 21
- Common Fields and Classes .................................................. 23
  - Member Types ................................................................. 23
  - Name-Value Pair ............................................................ 27
  - Status ............................................................................. 27
- Resource Hierarchy Classes .................................................... 28
  - Entry ............................................................................... 29
  - Item ................................................................................ 30
  - Dimension Item .............................................................. 31
  - Member .......................................................................... 32
  - Dimension Member .......................................................... 33
  - System Member ............................................................... 33
- Other Common Classes ........................................................... 34
  - Resource Meta ................................................................. 34
  - Resource Field Meta ........................................................ 35
  - Resource Key .................................................................. 36
  - Resource Audit ................................................................. 36
  - Request Result ................................................................. 37
  - Resource Fault Detail ....................................................... 38
  - Additional Data ............................................................... 38
  - Page Info ......................................................................... 38
  - Resource Audit Results .................................................... 39
- Pkey Map Classes ..................................................................... 40
  - Resource Hierarchy Including Pkey Maps ............................ 40
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provisionable Remote Resource Pkey Map Types</td>
<td>40</td>
</tr>
<tr>
<td>Non Provisionable Remote Resource Pkey Map Types</td>
<td>42</td>
</tr>
<tr>
<td>Member Pkey Map Types</td>
<td>44</td>
</tr>
<tr>
<td>Pkey Map Class</td>
<td>45</td>
</tr>
<tr>
<td>Item Pkey Class</td>
<td>46</td>
</tr>
<tr>
<td>Member Pkey Class</td>
<td>46</td>
</tr>
<tr>
<td>Using the Pkey Map Classes</td>
<td>47</td>
</tr>
<tr>
<td>Field Validation</td>
<td>47</td>
</tr>
<tr>
<td><strong>System Resource Types</strong></td>
<td>48</td>
</tr>
<tr>
<td>Folder Resource Item</td>
<td>48</td>
</tr>
<tr>
<td>Description</td>
<td>48</td>
</tr>
<tr>
<td>Fields</td>
<td>48</td>
</tr>
<tr>
<td>Model</td>
<td>49</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>50</td>
</tr>
<tr>
<td>Group Resource Item</td>
<td>50</td>
</tr>
<tr>
<td>Description</td>
<td>50</td>
</tr>
<tr>
<td>Fields</td>
<td>50</td>
</tr>
<tr>
<td>Model</td>
<td>51</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>51</td>
</tr>
<tr>
<td>User Resource Item</td>
<td>52</td>
</tr>
<tr>
<td>Description</td>
<td>52</td>
</tr>
<tr>
<td>Fields</td>
<td>52</td>
</tr>
<tr>
<td>Model</td>
<td>54</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>54</td>
</tr>
<tr>
<td><strong>Provisionable Resource Types</strong></td>
<td>55</td>
</tr>
<tr>
<td>Agent Resource Item</td>
<td>55</td>
</tr>
<tr>
<td>Description</td>
<td>55</td>
</tr>
<tr>
<td>Fields</td>
<td>55</td>
</tr>
<tr>
<td>Model</td>
<td>58</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>58</td>
</tr>
<tr>
<td>Agent Desktop Resource Item</td>
<td>59</td>
</tr>
<tr>
<td>Description</td>
<td>59</td>
</tr>
<tr>
<td>Fields</td>
<td>59</td>
</tr>
<tr>
<td>Model</td>
<td>63</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>63</td>
</tr>
<tr>
<td>Agent Team Resource Item</td>
<td>64</td>
</tr>
<tr>
<td>Description</td>
<td>64</td>
</tr>
<tr>
<td>Fields</td>
<td>64</td>
</tr>
<tr>
<td>Model</td>
<td>65</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>66</td>
</tr>
<tr>
<td>Call Type Resource Item</td>
<td>66</td>
</tr>
<tr>
<td>Description</td>
<td>66</td>
</tr>
<tr>
<td>Fields</td>
<td>66</td>
</tr>
<tr>
<td>Model</td>
<td>68</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>68</td>
</tr>
<tr>
<td>Calling Search Space Resource Item</td>
<td>69</td>
</tr>
<tr>
<td>Description</td>
<td>69</td>
</tr>
<tr>
<td>Fields</td>
<td>69</td>
</tr>
<tr>
<td>Model</td>
<td>70</td>
</tr>
<tr>
<td>Resource Item</td>
<td>Page</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>REST Protocol</td>
<td>70</td>
</tr>
<tr>
<td>Department Resource Item</td>
<td>71</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>71</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>71</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>72</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>72</td>
</tr>
<tr>
<td>Device Profile Resource Item</td>
<td>73</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>73</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>73</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>74</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>74</td>
</tr>
<tr>
<td>Dialed Number Resource Item</td>
<td>75</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>75</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>75</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>76</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>76</td>
</tr>
<tr>
<td>Directory Number Resource Item</td>
<td>77</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>77</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>77</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>79</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>79</td>
</tr>
<tr>
<td>Enterprise Skillgroup Resource Item</td>
<td>80</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>80</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>80</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>81</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>81</td>
</tr>
<tr>
<td>Expanded Call Variable Resource Item</td>
<td>82</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>82</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>82</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>83</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>83</td>
</tr>
<tr>
<td>IP Endpoint Resource Item</td>
<td>84</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>84</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>84</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>86</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>86</td>
</tr>
<tr>
<td>.............................. Auto Selection of Device Pool</td>
<td>87</td>
</tr>
<tr>
<td>Label Resource Item</td>
<td>88</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>88</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>88</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>89</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>89</td>
</tr>
<tr>
<td>Network VRU Script Resource Item</td>
<td>90</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>90</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>90</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>91</td>
</tr>
<tr>
<td>.............................. REST Protocol</td>
<td>91</td>
</tr>
<tr>
<td>Person Resource Item</td>
<td>92</td>
</tr>
<tr>
<td>.............................. Description</td>
<td>92</td>
</tr>
<tr>
<td>.............................. Fields</td>
<td>92</td>
</tr>
<tr>
<td>.............................. Model</td>
<td>94</td>
</tr>
<tr>
<td>Service</td>
<td>Page</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>178</td>
</tr>
<tr>
<td>Delete Agent</td>
<td>178</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>178</td>
</tr>
<tr>
<td>Retrieve Agent</td>
<td>179</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>179</td>
</tr>
<tr>
<td>Search For Agent</td>
<td>179</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>179</td>
</tr>
<tr>
<td>Describe Agent and Person</td>
<td>180</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>180</td>
</tr>
<tr>
<td>Create Agent Team</td>
<td>180</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>180</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>181</td>
</tr>
<tr>
<td>Update Agent Team</td>
<td>182</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>182</td>
</tr>
<tr>
<td>Add/Remove Agent Team Members</td>
<td>183</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>183</td>
</tr>
<tr>
<td>Create Call Type</td>
<td>183</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>183</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>184</td>
</tr>
<tr>
<td>Update Call Type</td>
<td>185</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>185</td>
</tr>
<tr>
<td>Add/Remove Routing Script Members</td>
<td>186</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>186</td>
</tr>
<tr>
<td>Create Dialed Number</td>
<td>186</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>186</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>187</td>
</tr>
<tr>
<td>Update Dialed Number</td>
<td>188</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>188</td>
</tr>
<tr>
<td>Create Directory Number</td>
<td>189</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>189</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>190</td>
</tr>
<tr>
<td>Update Directory Number</td>
<td>191</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>191</td>
</tr>
<tr>
<td>Create Folder</td>
<td>191</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>191</td>
</tr>
<tr>
<td>Update Folder</td>
<td>192</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>192</td>
</tr>
<tr>
<td>Create Group</td>
<td>194</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>194</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>195</td>
</tr>
<tr>
<td>Update Group</td>
<td>196</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>196</td>
</tr>
<tr>
<td>Create IP Endpoint</td>
<td>196</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>196</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>197</td>
</tr>
<tr>
<td>Update IP Endpoint</td>
<td>199</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>199</td>
</tr>
<tr>
<td>Create Person</td>
<td>199</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>199</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>200</td>
</tr>
<tr>
<td>Update Person</td>
<td>201</td>
</tr>
</tbody>
</table>
Use Case Sequence ............................................................... 201
Create Skill Group................................................................. 201
    Use Case Sequence.......................................................... 201
    SOAP Example................................................................. 202
Update Skill Group............................................................... 203
    Use Case Sequence.......................................................... 203
Create Tenant ........................................................................ 204
    Use Case Sequence.......................................................... 204
Update Tenant ........................................................................ 205
    Use Case Sequence.......................................................... 205
Create User ........................................................................... 205
    Use Case Sequence.......................................................... 205
    SOAP Example................................................................. 206
Update User ........................................................................... 207
    Use Case Sequence.......................................................... 207
Agent to Skill Group Membership ............................................. 208
    SOAP Example................................................................. 208
Agent to Agent Team Membership .......................................... 209
    SOAP Example................................................................. 209
IP Endpoint to Directory Number Membership ....................... 211
    SOAP Example................................................................. 211
User to Group Membership ..................................................... 213
    SOAP Example................................................................. 213
Group to Group Membership .................................................. 214
    SOAP Example................................................................. 214
Dialed Number to Call Type Membership .................................. 215
    SOAP Example................................................................. 215
Agent Re-Skilling .................................................................... 218
User Creation .......................................................................... 220
Move Unallocated Dialed Number to Tenant Folder .................. 221
Link Dialed Number to Call Type/Routing Script ...................... 222
Unlink Dialed Number to Call Type/Routing Script .................... 223
Rename Resources .................................................................... 224
Create Agent with Pkey Maps, Method 1 (Explicit Pkey Maps) ... 225
    Description....................................................................... 225
    Use Case Sequence.......................................................... 225
    SOAP Example................................................................. 225
Create Agent Team with Pkey Maps, Method 2 (Implicit Pkey Maps) ... 228
    Description....................................................................... 228
    Use Case Sequence.......................................................... 228
    SOAP Example................................................................. 228
Update Agent (Additional Pkey Map) .................................... 229
    Description....................................................................... 229
    Use Case Sequence.......................................................... 229
    SOAP Example................................................................. 230
Update Agent (Delete a Pkey Map) ........................................ 231
    Description....................................................................... 231
    Use Case Sequence.......................................................... 231
    SOAP Example................................................................. 232
Create Agent To Agent Team Membership (Two Remote Equipment Mappings) ................................................. 232
<table>
<thead>
<tr>
<th>Description</th>
<th>232</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Sequence</td>
<td>233</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>233</td>
</tr>
<tr>
<td>Create Precision Attribute</td>
<td>235</td>
</tr>
<tr>
<td>Soap Example</td>
<td>235</td>
</tr>
<tr>
<td>Create Precision Queue</td>
<td>236</td>
</tr>
<tr>
<td>Description</td>
<td>236</td>
</tr>
<tr>
<td>Use Case Sequence</td>
<td>237</td>
</tr>
<tr>
<td>SOAP Example</td>
<td>237</td>
</tr>
</tbody>
</table>

| Update Precision Queue (Add Precision Queue Step With Two Precision Attributes) | 240 |
| Description | 240 |
| SOAP Example | 240 |
| Update Precision Queue (Delete Precision Queue Step) | 244 |
| Description | 244 |
| SOAP Example | 244 |

### 4. Subscriptions Web Service

<table>
<thead>
<tr>
<th>Overview</th>
<th>245</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>245</td>
</tr>
<tr>
<td>Specification</td>
<td>245</td>
</tr>
<tr>
<td>SOAP</td>
<td>245</td>
</tr>
<tr>
<td>REST</td>
<td>245</td>
</tr>
<tr>
<td>Securing Notifications with SSL</td>
<td>245</td>
</tr>
<tr>
<td>Introduction</td>
<td>245</td>
</tr>
<tr>
<td>Obtain and Install a Digital Certificate for the Unified CCMP Server</td>
<td>246</td>
</tr>
<tr>
<td>Obtain and Install a Digital Certificate for the Subscriber Server</td>
<td>247</td>
</tr>
<tr>
<td>Configure the Unified CCMP Server Endpoint Behaviour</td>
<td>247</td>
</tr>
<tr>
<td>Install the Public Key of the Subscriber Server Certificate onto the Unified CCP Server</td>
<td>249</td>
</tr>
<tr>
<td>Install the Root CA Certificate on the Subscriber server</td>
<td>249</td>
</tr>
<tr>
<td>Restart the Services</td>
<td>249</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subscriptions Web Service</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subscribe</td>
<td>250</td>
</tr>
<tr>
<td>Description</td>
<td>250</td>
</tr>
<tr>
<td>Example</td>
<td>250</td>
</tr>
<tr>
<td>Get Status</td>
<td>251</td>
</tr>
<tr>
<td>Description</td>
<td>251</td>
</tr>
<tr>
<td>Example</td>
<td>251</td>
</tr>
<tr>
<td>Renew</td>
<td>252</td>
</tr>
<tr>
<td>Description</td>
<td>252</td>
</tr>
<tr>
<td>Example</td>
<td>253</td>
</tr>
<tr>
<td>Unsubscribe</td>
<td>253</td>
</tr>
<tr>
<td>Description</td>
<td>253</td>
</tr>
<tr>
<td>Example</td>
<td>254</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>More Information about Subscriptions and Notifications</th>
<th>254</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subscribing for Notifications</td>
<td>254</td>
</tr>
</tbody>
</table>
Preface

Purpose

This document describes the Web Service APIs that CCMP exposes and explains how to use them. The Web Services APIs allow resource and hierarchy management operations to be performed remotely from a third party client application. The Web Services APIs also allow the client to subscribe to and receive notifications of state changes in contact center resources.

Audience

This document is intended for web developers who are writing applications that need to use the CCMP Web Service APIs. The reader should be familiar with web service integration, and also have a basic understanding of CCMP and how it interacts with the contact center environment.

Organization

The sections of this document are as follows:

Chapter 1  Introduction to Web Services

This section contains an introduction to CCMP and the Web Service APIs

Chapter 2  Common Data Types

This section defines some data types that are common to more than one Web Service

Chapter 3  Resource Management Web Service

This section describes the data types and methods in Resource Management Web Service APIs, and contains example code and scenarios showing how to use the Resource Management Web Service.

Chapter 4  Subscriptions Web Service

This section describes the data types and methods in Subscriptions Web Service APIs, and contains example code and scenarios showing how to use the Subscriptions Web Service.
Terms and Definitions

For a comprehensive description of terms relating to Unified CCMP and computer telephony integration, please refer to the Glossary section in the online help.

Some of the terms and definitions used in this document are given in the table below.

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTI</td>
<td>Computer telephony integration.</td>
</tr>
<tr>
<td>Dimension</td>
<td>A BI term for a category. In this context is another term for resource (see Resource).</td>
</tr>
<tr>
<td>Delete Operation</td>
<td>A delete operation is the removal of a resource from the remote system. If the resource is a child resource then the remote system may allow flagging it as deleted. Unified CCMP uses effective dating to allow all resources to be flagged in this manner.</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>A parent/child data structure that allows for deep partitioning schemes modeled on a variety of user defined requirements for example, geographical, business unit, reseller etc.</td>
</tr>
<tr>
<td>Meta-model</td>
<td>Unified CCMP data system that holds the past, current and planned state of the entire contact center.</td>
</tr>
<tr>
<td>Move Operation</td>
<td>The moving of a resource in the resource hierarchy. May trigger a type-2 operation (generating a new resource identifier) if the resource is moved between tenants.</td>
</tr>
<tr>
<td>Multi-tenanted</td>
<td>A secure partitioning architecture that allows a single platform to support multiple customers.</td>
</tr>
</tbody>
</table>
### Product Naming Conventions

In this release, the product names defined in the table below have changed. The New Name (long version) is reserved for the first instance of that product name and in all headings. The New Name (short version) is used for subsequent instances of the product name.

**Note** This document uses the naming conventions provided in each GUI, which means that in some cases the old product name is in use.

<table>
<thead>
<tr>
<th>Old Product Name</th>
<th>New Name (long version)</th>
<th>New Name (short version)</th>
</tr>
</thead>
</table>

---

**Term** | **Definition**  
Purge | A purge operation is the forced removal of a resource from the remote system and may require cascade delete operations.  
Resource | An item of configuration data such as agent, dialed number etc.  
Resource Prefix | An alphanumeric identifier which is a part of the unique identifier of the resource.  
System Identifier | A unique name, not the numeric primary key, that is used to identify a resource. For example, in Unified CCE this would be the `EnterpriseName` property. Unified CCMP will typically create this field and ensure that it is unique across all distributed systems. An example naming convention is `nnnnMMMaa`, where:  
  `nnnn` = tenant identifier,  
  `MMM` = equipment instance identifier,  
  `aa` = resource identifier.  
Tenant | A specialized node within the multi-tenanted folder structure which represents a customer instance that has telephony isolation from other tenants.  
Type-2 Effective Dating | A BI technique in which every item of configuration data has a start date and end date to allow its life-time to be strictly controlled.  
User | In this context, is defined as a person using Unified CCMP or the monitored equipment, optionally may also be an Agent.
<table>
<thead>
<tr>
<th>Old Product Name</th>
<th>New Name (long version)</th>
<th>New Name (short version)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IPCC Enterprise Edition</td>
<td>Cisco Unified Contact Center Enterprise</td>
<td>Unified CCE</td>
</tr>
<tr>
<td>Cisco IPCC Hosted Edition</td>
<td>Cisco Unified Contact Center Hosted</td>
<td>Unified CCH</td>
</tr>
<tr>
<td>Cisco Intelligent Contact Management (ICM) Enterprise Edition</td>
<td>Cisco Unified Intelligent Contact Management (ICM) Enterprise</td>
<td>Unified ICM</td>
</tr>
<tr>
<td>Cisco Intelligent Contact Management (ICM) Hosted Edition</td>
<td>Cisco Unified Intelligent Contact Management (ICM) Hosted</td>
<td></td>
</tr>
<tr>
<td>Cisco CallManager/Cisco Unified CallManager</td>
<td>Cisco Unified Communications Manager</td>
<td>Unified CM</td>
</tr>
</tbody>
</table>

**Conventions**

This document uses the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface font is used to indicate commands, such as entries, keys, buttons, folders and submenu names. For example:</td>
</tr>
<tr>
<td></td>
<td>• Choose <strong>Edit &gt; Find</strong></td>
</tr>
<tr>
<td></td>
<td>• Click <strong>Finish</strong></td>
</tr>
<tr>
<td><strong>italic</strong></td>
<td>Italic font is used to indicate the following:</td>
</tr>
<tr>
<td></td>
<td>• To introduce a new term; for example: A <em>skill group</em> is a collection of agents who share similar skills</td>
</tr>
<tr>
<td></td>
<td>• For emphasis; for example: <em>Do not</em> use the numerical naming convention</td>
</tr>
<tr>
<td></td>
<td>• A syntax value that the user must replace; for example: IF (<em>condition,</em> <em>true-value,</em> <em>false-value</em>)</td>
</tr>
<tr>
<td></td>
<td>• A book title; for example: Refer to the <em>Cisco CRS Installation Guide</em></td>
</tr>
<tr>
<td><strong>window</strong></td>
<td>Window font, such as Courier, is used for the following:</td>
</tr>
<tr>
<td></td>
<td>• Text as it appears in code or that the window displays; for example: <em>&lt;html&gt;&lt;title&gt;Cisco Systems, Inc. &lt;/title&gt;&lt;/html&gt;</em></td>
</tr>
</tbody>
</table>
Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see What’s New in Cisco Product Documentation, at:


Subscribe to What’s New in Cisco Product Documentation, which lists all new and revised Cisco technical documentation as an RSS feed and delivers content directly to your desktop using a reader application. The RSS feeds are a free service.

Field Alerts and Field Notices

Cisco products may be modified or key processes may be determined to be important. These are announced through use of the Cisco Field Alerts and Cisco Field Notices. You can register to receive Field Alerts and Field Notices through the Product Alert Tool on Cisco.com. This tool enables you to create a profile to receive announcements by selecting all products of interest.

Log into www.cisco.com and then access the tool at http://www.cisco.com/cisco/support/notifications.html

Documentation Feedback

To provide comments about this document, send an email message to the following address: contactcenterproducts_docfeedback@cisco.com

We appreciate your comments.
1. Introduction to Web Services

Unified CCMP Overview

The Unified CCMP product suite is a shared schema multi-tenanted platform that supports
• Provisioning
• Service Management.

Unified CCMP exposes Web Services APIs that allow third party client applications to:
• perform resource management operations
• request and receive notifications when resources change state.

This document describes the Web Services APIs that Unified CCMP exposes and how to use them.

Table 1-1 Unified CCMP Web Services

<table>
<thead>
<tr>
<th>Web Service</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Management</td>
<td>Allows the client to provision and manage contact center resources.</td>
</tr>
<tr>
<td>Subscriptions</td>
<td>Allows the client to subscribe to notifications about state changes to contact center resources.</td>
</tr>
</tbody>
</table>

Web Service Protocols

The Unified CCMP Web Services support two protocols:
• Simple Object Access Protocol, or SOAP (see http://en.wikipedia.org/wiki/SOAP)
• Representational state transfer, or REST (see http://en.wikipedia.org/wiki/REST)

Typically, inter-platform interactions, such as an Enterprise Service Bus, use SOAP because of its excellent standards-based interoperability, ease of multi-firewall traversal and existing engineering skillsets, whereas thin client and mobile applications typically use REST because of its low footprint and ease of coding.

All the Unified CCMP Web Services support SOAP as a default protocol. In addition, the Resource Management Web Service also supports the REST protocol with the same payload types (using both XML and JSON content types). The SOAP
implementation is also built on architectural REST principles but within the SOAP envelope.

This simultaneous support of both SOAP and REST allows for maximum interoperability and extends the traditional Unified CCMP API use to client gadgets.

The following table shows the typical usages of the Unified CCMP Web Services.

<table>
<thead>
<tr>
<th>Web Service</th>
<th>SOAP</th>
<th>REST XML</th>
<th>REST JSON</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Management</td>
<td>Enterprise Service Bus / Thick client</td>
<td>Java or other thin client</td>
<td>JavaScript gadgets/widgets and mobile</td>
</tr>
<tr>
<td>Subscriptions</td>
<td>Enterprise Service Bus/ Thick client</td>
<td>N/A, SOAP only concept</td>
<td>N/A, SOAP only concept</td>
</tr>
</tbody>
</table>

**SOAP**

The Unified CCMP SOAP web service design concepts are:

- The technology base is built on a SOAP 1.2 web service stack to address the widest market at present.
- For all Unified CCMP Web Services, the WS-Security and WS-Addressing standards are implemented.
- The WS-Eventing and Subscription services have been implemented to allow for asynchronous notification of changes of resources.
- The Unified CCMP Resource Management Web Service has combined elements of WS-Resourcing and WS-Management (these could not be used directly since both contain proprietary elements).

**REST**

The provisioning resources are organized into collections to allow clients and the Unified CCMP server to perform queries on the collections and to use the collection as a factory for creating new resources. Since they are collection resources they are plural nouns with a URI path that indicates the hierarchy. Unified CCMP supports three types of resource addressing:

1. Classic resource addressing. This is the usual REST collection resource paradigm where one or more resources of the same type may be contained in a path. The URI is hierarchical and contains:
   - the resource type as part of the path
   - a single identifier to get, put or delete a resource of that type.
   For example

   ```
   GET /resources/agents/1006
   ```

   where 1006 is the entity id and agents is the type of resource.
2. Composite resource collection. This is a collection URI where the identifier is a composite key made up of the entity id and the entity type. Resources of all types may be addressed under a common resources path. This is typically used where provisioning resources of all types may be held.

   For example

   GET /resources/1006, Agent

3. Anonymous collections. These are an extension of both classic and composite resource addressing and use multiple ids to provide bulk operation capability. The key to understanding this extension is that a sub-collection of resources is itself a collection which can be cached. The sub-collection identifier is the concatenation of entity ids (and possible types). Both classic and composite addressing modes are supported:

   For example

   GET /resources/agents/1006, 1007, 1008

   GET /resources/1006, Agent | 3412, Label

<table>
<thead>
<tr>
<th>REST Verb</th>
<th>Create</th>
<th>Update</th>
<th>Describe / Search</th>
<th>Retrieve</th>
<th>Delete</th>
<th>Audit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collection Type:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classic Addressing</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>resource/{type}/{id}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Composite Addressing</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>resource/{id,type}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Batch Classic Addressing</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>resource/{type}/{id}, {id}, {id}...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Batch Composite Addressing</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>resource/{id,type}, {id,type}...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Security

Resource Management and Subscriptions Web Services

The Resource Management and Subscriptions Web Services use Unified CCMP’s role-based access control system to secure and restrict what the client can do or see. The authentication mode depends on the protocol being used.
The Resource Management and Subscriptions Web Services are secured using WS-Security username tokens. You must pass a valid Unified CCMP username and password to access the service methods.

**SOAP**

You can use the SoapUI tool, *Project Settings* window, *Security Configuration* tab to configure a security profile. This profile can then be selected for each request using the pop-up *Auth* tab on the request editor window.

Figure 1-1 shows how to configure a security profile using SOAP UI.

*Figure 1-1 Configuring a Security profile using SOAP UI*

![Figure 1-1 Configuring a Security profile using SOAP UI](image)

Figure 1-1 shows how to add a WS-Security section to the SOAP header sent to the Unified CCMP Server.
REST

The Resource Management Web Service uses Basic Authentication with the REST protocol. In this case, the client encodes the user name and password in Base-64 and adds it to the HTTP header.

An example is

```
GET /private/index.html HTTP/1.1
Host: local host
Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
```

The implementation does not support challenge and response and so the responses cannot be viewed using a browser.

Notifications

Notifications sent to clients who have subscribed for them using the Subscriptions Web Service are secured using mutual certificates.

For more information about configuring security for notifications, see, see section Securing Notifications with SSL.

More Information

For more details about the Unified CCMP security model, see the Security Guide for Cisco Unified Contact Center Management Portal.
2. Common Data Types

The Unified CCMP Web Service APIs allow third parties to perform various different operations on contact center resources and their associated metadata.

This section describes the common data types and the actions that can be performed on them.

Remote Resource Types

These types correspond directly to the remote resources in the Contact Center environment in which Unified CCMP operates. Some of these types may also be used with the Resource Management Web Service APIs to allow client utilities and a mid-tier interface to perform adds, updates and deletes.

Provisionable Remote Resource Types

These remote resource types are fully supported by the Resource Management Web Services.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Internal Name</th>
<th>REST Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>IT_AGENT</td>
<td>agent</td>
</tr>
<tr>
<td>Agent Desktop</td>
<td>IT_AGENT_DESKTOP</td>
<td>agent - desktop</td>
</tr>
<tr>
<td>Agent Team</td>
<td>IT_AGENT_TEAM</td>
<td>agent - team</td>
</tr>
<tr>
<td>Call Type</td>
<td>IT_CALL_TYPE</td>
<td>call - type</td>
</tr>
<tr>
<td>Calling Search Space</td>
<td>IT_CALLING_SEARCH_SPACE</td>
<td>calling-search-space</td>
</tr>
<tr>
<td>Department</td>
<td>IT_DEPARTMENT</td>
<td>department</td>
</tr>
<tr>
<td>Device Profile</td>
<td>IT_DEVICE_PROFILE</td>
<td>device-profile</td>
</tr>
<tr>
<td>Dialed Number</td>
<td>IT_DIALED_NUMBER</td>
<td>dialed-number</td>
</tr>
<tr>
<td>Directory Number</td>
<td>IT_DIRECTORY_NUMBER</td>
<td>directory-number</td>
</tr>
<tr>
<td>Enterprise Skill Group</td>
<td>IT_ENTERPRISE_SKILLGROUP</td>
<td>enterprise-skill-group</td>
</tr>
<tr>
<td>Expanded Call Variable</td>
<td>IT_EXPANDED_CALL_VARIABLE</td>
<td>expanded-call-variable</td>
</tr>
<tr>
<td>IP Endpoint</td>
<td>IT_IP_ENDPOINT</td>
<td>ip-endpoint</td>
</tr>
</tbody>
</table>
### Remote Resource Types

#### Resource Type

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Internal Name</th>
<th>REST Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>IVR Script (or VXML Application)</td>
<td>IT_IVR_SCRIPT</td>
<td>ivr-script</td>
</tr>
<tr>
<td>Label</td>
<td>IT_LABEL</td>
<td>label</td>
</tr>
<tr>
<td>Media file</td>
<td>IT_MEDIAFILE</td>
<td>mediafile</td>
</tr>
<tr>
<td>Network VRU Script</td>
<td>IT_NETWORK_VRU_SCRIPT</td>
<td>network-vru-script</td>
</tr>
<tr>
<td>Person</td>
<td>IT_PERSON</td>
<td>person</td>
</tr>
<tr>
<td>Precision Attribute</td>
<td>IT_PRECISION_ATTRIBUTE</td>
<td>precision-attribute</td>
</tr>
<tr>
<td>Precision Queue</td>
<td>IT_PRECISION_QUEUE</td>
<td>precision-queue</td>
</tr>
<tr>
<td>Precision Queue Step</td>
<td>IT_PRECISION_QUEUE_STEP</td>
<td>precision-queue-step</td>
</tr>
<tr>
<td>Route</td>
<td>IT_ROUTE</td>
<td>route</td>
</tr>
<tr>
<td>Route Partition</td>
<td>IT_ROUTE_PARTITION</td>
<td>route-partition</td>
</tr>
<tr>
<td>Service</td>
<td>IT_SERVICE</td>
<td>service</td>
</tr>
<tr>
<td>Skill Group</td>
<td>IT_SKILLGROUP</td>
<td>skill-group</td>
</tr>
<tr>
<td>Tenant</td>
<td>IT_TENANT</td>
<td>tenant</td>
</tr>
<tr>
<td>User Variable</td>
<td>IT_USER_VARIABLE</td>
<td>user-variable</td>
</tr>
</tbody>
</table>

#### Non Provisionable Remote Resource Types

These remote resource types are supported by the Resource Management Web Services for searching only.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Internal Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Announcement</td>
<td>IT_ANNOUNCEMENT</td>
</tr>
<tr>
<td>Application Gateway</td>
<td>IT_APPLICATION_GATEWAY</td>
</tr>
<tr>
<td>Application Instance</td>
<td>IT_APPLICATION_INSTANCE</td>
</tr>
<tr>
<td>Bucket Interval</td>
<td>IT_BUCKET_INTERVAL</td>
</tr>
<tr>
<td>Call Manager Group</td>
<td>IT_CALL_MANAGER_GROUP</td>
</tr>
<tr>
<td>Call Source</td>
<td>IT_CALL_SOURCE</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Internal Name</td>
</tr>
<tr>
<td>-------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>Campaign</td>
<td>IT_CAMPAIGN</td>
</tr>
<tr>
<td>Category</td>
<td>ITCATEGORY</td>
</tr>
<tr>
<td>Chargeband</td>
<td>IT_CHARGEBAND</td>
</tr>
<tr>
<td>Cli</td>
<td>IT_CLI</td>
</tr>
<tr>
<td>Custom Entry</td>
<td>IT_CUSTOM_ENTRY</td>
</tr>
<tr>
<td>Date Time Setting</td>
<td>IT_DATE_TIME_SETTING</td>
</tr>
<tr>
<td>Device Pool</td>
<td>IT_DEVICE_POOL</td>
</tr>
<tr>
<td>Device Target</td>
<td>IT_DEVICE_TARGET</td>
</tr>
<tr>
<td>Dialer</td>
<td>IT_DIALER</td>
</tr>
<tr>
<td>Dial Number Plan</td>
<td>IT.Dial_NUMBER_PLAN</td>
</tr>
<tr>
<td>Enterprise Route</td>
<td>IT_ENTERPRISE_ROUTE</td>
</tr>
<tr>
<td>Enterprise Service</td>
<td>IT_ENTERPRISE_SERVICE</td>
</tr>
<tr>
<td>Gateway Function</td>
<td>IT_GATEWAY_FUNCTION</td>
</tr>
<tr>
<td>Gateway Result</td>
<td>IT_GATEWAY_RESULT</td>
</tr>
<tr>
<td>Gateway Server</td>
<td>IT_GATEWAY_SERVER</td>
</tr>
<tr>
<td>ICR Instance</td>
<td>IT_ICR_INSTANCE</td>
</tr>
<tr>
<td>Import Rule</td>
<td>IT_IMPORT_RULE</td>
</tr>
<tr>
<td>IP Endpoint Button Template</td>
<td>IT.IP_ENDPOINT_BUTTON_Template</td>
</tr>
<tr>
<td>IP Endpoint Model</td>
<td>IT.IP_ENDPOINT_MODEL</td>
</tr>
<tr>
<td>IVR Entry Point</td>
<td>IT_IVR_ENTRY_POINT</td>
</tr>
<tr>
<td>IVR Module</td>
<td>IT_IVR_MODULE</td>
</tr>
<tr>
<td>IVR Routing Target</td>
<td>IT_IVR_ROUTING_TARGET</td>
</tr>
<tr>
<td>IVR Script Node</td>
<td>IT_IVR_SCRIPT_NODE</td>
</tr>
<tr>
<td>Logical Interface Controller</td>
<td>IT_LOGICAL_INTERFACE_CONTROLLER</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Internal Name</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Media Class</td>
<td>IT_MEDIA_CLASS</td>
</tr>
<tr>
<td>Media File Server</td>
<td>IT_MEDIAFILE_SERVER</td>
</tr>
<tr>
<td>Media Routing Domain</td>
<td>IT_MEDIA_ROUTING_DOMAIN</td>
</tr>
<tr>
<td>Network Trunk Group</td>
<td>IT_NETWORK_TRUNK_GROUP</td>
</tr>
<tr>
<td>Network Vru</td>
<td>IT_NETWORK_VRU</td>
</tr>
<tr>
<td>Object Type</td>
<td>IT_OBJECT_TYPE</td>
</tr>
<tr>
<td>Peripheral</td>
<td>IT_PERIPHERAL</td>
</tr>
<tr>
<td>Physical Interface Controller</td>
<td>IT_PHYSICAL_INTERFACE_CONTROLLER</td>
</tr>
<tr>
<td>Port</td>
<td>IT_PORT</td>
</tr>
<tr>
<td>Query Rule</td>
<td>IT_QUERY_RULE</td>
</tr>
<tr>
<td>Rating Period</td>
<td>IT_RATING_PERIOD</td>
</tr>
<tr>
<td>Reason Code</td>
<td>IT_REASON_CODE</td>
</tr>
<tr>
<td>Region</td>
<td>IT_REGION</td>
</tr>
<tr>
<td>Route Partition</td>
<td>IT_ROUTE_PARTITION</td>
</tr>
<tr>
<td>Routing Client</td>
<td>IT_ROUTING_CLIENT</td>
</tr>
<tr>
<td>Routing Script</td>
<td>IT_ROUTING_SCRIPT</td>
</tr>
<tr>
<td>Schedule</td>
<td>IT_SCHEDULE</td>
</tr>
<tr>
<td>Scheduled Target</td>
<td>IT_SCHEDULED_TARGET</td>
</tr>
<tr>
<td>Script</td>
<td>IT_SCRIPT</td>
</tr>
<tr>
<td>Script Node</td>
<td>IT_SCRIPT_NODE</td>
</tr>
<tr>
<td>Strategy</td>
<td>IT_STRATEGY</td>
</tr>
<tr>
<td>Timeband</td>
<td>IT_TIMEBAND</td>
</tr>
<tr>
<td>Tli</td>
<td>IT_TLI</td>
</tr>
<tr>
<td>Trunk</td>
<td>IT_TRUNK</td>
</tr>
</tbody>
</table>
Chapter 2: Common Data Types

System Resource Types

System Resource types are contained in the Unified CCMP Database. They are not provisioned to remote contact center equipment, but are Unified CCMP-specific items used for organizing and accessing other resource types.

Provisionable System Resource Types

These system resource types are fully supported by the Resource Management Web Services.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Internal Name</th>
<th>REST Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>IT_USER</td>
<td>user</td>
</tr>
<tr>
<td>Group</td>
<td>IT_GROUP</td>
<td>group</td>
</tr>
<tr>
<td>Folder</td>
<td>IT_FOLDER</td>
<td>folder</td>
</tr>
</tbody>
</table>

Non Provisionable System Resource Types

These system resource types are supported by the Resource Management Web Services for searching only.

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Internal Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>IT_DATE</td>
</tr>
<tr>
<td>NNG</td>
<td>IT_NNG</td>
</tr>
<tr>
<td>Time</td>
<td>IT_TIME</td>
</tr>
<tr>
<td>Cluster Connection</td>
<td>IT_CLUSTER_CONNECTION</td>
</tr>
<tr>
<td>Cluster Connection Type</td>
<td>IT_CLUSTER_CONNECTION_TYPE</td>
</tr>
<tr>
<td>Cluster Resource</td>
<td>IT_CLUSTER_RESOURCE</td>
</tr>
<tr>
<td>Data Type</td>
<td>Internal Name</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>Cluster Resource Instance</td>
<td>IT_CLUSTERRESOURCE_INSTANCE</td>
</tr>
<tr>
<td>Cluster Resource Instance Component</td>
<td>IT_CLUSTERRESOURCEINSTANCECOMPONENT</td>
</tr>
<tr>
<td>Cluster Resource Instance Component Type</td>
<td>IT_CLUSTERRESOURCEINSTANCECOMPONENTTYPE</td>
</tr>
<tr>
<td>Cluster Resource Type</td>
<td>IT_CLUSTERRESOURCETYPE</td>
</tr>
<tr>
<td>Cluster Resource Type Group</td>
<td>IT_CLUSTERRESOURCETYPEGROUP</td>
</tr>
<tr>
<td>Cluster Server</td>
<td>IT_CLUSTERSERVER</td>
</tr>
<tr>
<td>Connected System</td>
<td>IT_CONNECTEDSYSTEM</td>
</tr>
<tr>
<td>Custom 0</td>
<td>IT_CUSTOM0</td>
</tr>
<tr>
<td>Custom 1</td>
<td>IT_CUSTOM1</td>
</tr>
<tr>
<td>Custom 2</td>
<td>IT_CUSTOM2</td>
</tr>
<tr>
<td>Custom 3</td>
<td>IT_CUSTOM3</td>
</tr>
<tr>
<td>Custom 4</td>
<td>IT_CUSTOM4</td>
</tr>
<tr>
<td>Custom 5</td>
<td>IT_CUSTOM5</td>
</tr>
<tr>
<td>Custom 6</td>
<td>IT_CUSTOM6</td>
</tr>
<tr>
<td>Custom 7</td>
<td>IT_CUSTOM7</td>
</tr>
<tr>
<td>Custom 8</td>
<td>IT_CUSTOM8</td>
</tr>
<tr>
<td>Custom 9</td>
<td>IT_CUSTOM9</td>
</tr>
<tr>
<td>Fileshare</td>
<td>IT_FILESHARE</td>
</tr>
<tr>
<td>Email</td>
<td>IT_EMAIL</td>
</tr>
<tr>
<td>Form Template</td>
<td>IT_FORMTEMPLATE</td>
</tr>
<tr>
<td>Managerial Role</td>
<td>IT_MANAGERIALROLE</td>
</tr>
<tr>
<td>Organisation</td>
<td>IT_ORGANISATION</td>
</tr>
<tr>
<td>Printer</td>
<td>IT_PRINTER</td>
</tr>
</tbody>
</table>
**SCD Effective Dating**

Remote resource types and their memberships are modeled as type-2 Slowly Changing Dimension (SCD) properties. The full resource history is maintained by creating new time-bounded records for each significant change (see http://en.wikipedia.org/wiki/Slowly_changing_dimension for more information).

- The Resource Management Web Service uses SCDs for memberships only, but not items. For instance, re-skilling an agent from one skill group to another will result in an SCD change for the agent-skillgroup member whereas changing the details of the agent or the skill group will not.

Item effective dating can be used to schedule a provisioning operation to occur in the future, that is, setting the `EffectiveFrom` field to be a future date.

The resource fields used to maintain SCD properties are `EffectiveFrom`, `EffectiveTo` and `Latest`. Together, these fields define the lifecycle of the resource items and memberships.

For example, a remote resource may have the following values.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Surrogate</th>
<th>EffectiveFrom</th>
<th>EffectiveTo</th>
<th>Latest</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>1000</td>
<td>01/01/2012</td>
<td>06/06/2079</td>
<td>1</td>
</tr>
</tbody>
</table>

After a SCD event the resource may have the following values.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Surrogate</th>
<th>EffectiveFrom</th>
<th>EffectiveTo</th>
<th>Latest</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>1000</td>
<td>01/01/2012</td>
<td>02/02/2012</td>
<td>0</td>
</tr>
<tr>
<td>1000</td>
<td>1001</td>
<td>02/02/2012</td>
<td>06/06/2079</td>
<td>1</td>
</tr>
</tbody>
</table>

The surrogate key field is used to track the change.
3. Resource Management Web Service

Overview

Description

This section describes the Resource Management Web Service APIs. This web service allows third party client applications to invoke provisioning operations on the underlying equipment and to create system resources.

Concepts

The Resource Management Web Service APIs have the following high level concepts:

- Provisioning operations are based on:
  - Resources, such as Agents and Call Types
  - Resource Memberships which define the relationship between these Resources.

- Most resources memberships are many-to-many associations. This model is loosely based on the Frameworx Shared Information/Data model (SID), with extensions to support full multi-tenancy and type-2 SCD lifecycle management. See http://www.tmforum.org/TMForumFrameworx/1911/home.html for more information about the Frameworx SID model.

- Resources are organized in a folder tree, where each top-level folder is either owned by a tenant or is one of the special folders (for example, /Unallocated, /Shared or /Search).

- Each resource is dated with a common header which uniquely identifies it so its lifecycle can be tracked. Type-2 SCD lifecycle tracking is used, so there is the concept of a latest resource and previous resources, each of which has its own identifier.

- Each resource or resource member has a common header which uniquely identifies the resource, and a loosely-coupled data structure body. This structure has two advantages:
  - New resource types can be added without needing to regenerate the WSDL (SOAP) or to change existing clients (REST).
  - Versioning can be performed at the resource or resource member level and not at the API level, which reduces upgrade costs.

- Transactions are coarse grained, and each create, update or retrieve web service call supports an array of up to 100 operations to minimize API chatter.
• Provisioning transactions may be synchronous or asynchronous:
  • Provisioning transactions for system resources (for example, a request to create a Folder) use a synchronous request and a synchronous response.
  • Provisioning transactions for remote resources (for example, a request to create an Agent) use a synchronous request and an asynchronous response for maximum scaling. This means that a remote resource will go through a Pending state (accepted, but queued) before it is successfully provisioned and enters the Ready state (exists on underlying equipment and can be used).
• A change to a remote resource or one of its associated members makes that resource unavailable for any other changes until that change has propagated through the underlying equipment. This avoids race conditions and other system complexities.
• Provisioning requests are built around a number of simple verbs; create, update, delete, save, move, describe, retrieve, search and audit.
• If the client is using the SOAP protocol, the client can subscribe to be notified whenever changes to resources are detected. This is particularly important when local native interfaces can be used to change the underlying equipment, thus rendering any overlaying data model as out-of-date.

Item Types
The Unified CCMP Resource Management Web Service API has a class hierarchy that represents the resources and memberships available on the system. The following item types are supported:
• provisionable remote resource types
• non-provisionable remote resource types
• provisionable system resource types
• non-provisionable system resource types
• membership types.

Pkey Map Item Types
The Unified CCMP Resource Management Web Service API also provides a pkey map class that links a remote resource with its corresponding resource on the remote equipment. For example, an Agent pkey map links the agent details in Unified CCMP with the agent details on the remote equipment.

The following resource types can have pkey maps:
• provisionable resources (for example, Agents, Agent Teams, Dialed Numbers)
• non-provisionable resources (for example, Announcements, Application Gateways, Bucket Intervals)
• member types (for example, Agent Agent Team Members, Agent Skill Group Members, Dialed Number Call Type Members)
Pkey maps can be used to link remote resources to multiple remote equipment instances with different details on each. For example, a single Agent resource may be located on both a Unified CCE and a Unified CM, with different login details on each. The agent pkey map links the Unified CCMP agent with the remote agent on each equipment instance and stores the different login details for each.

Pkey maps are also used to link resource memberships to the corresponding relationships on remote equipment. For example, if an Agent is in an Agent Team then the Agent Agent Team Member pkey map links the Unified CCMP membership with the corresponding agent to agent team relationship on the remote equipment.

If you do not need to map resources to multiple equipment instances then you can ignore the pkey map resource and member classes although they are still present. All the information about a single equipment mapping is available directly from the resource or member class.

The exception is the Describe() method, since, if a field may be specified in a pkey map, the metadata about that field is returned against the pkey map type, not the corresponding resource type.

System resources (for example, Folders, Users, Groups) do not have pkey maps, as they do not exist on the remote equipment.

**Methods**

The Unified CCMP Resource Management Web Service API exposes the following methods:

- Create()
- Update()
- Delete()
- Save()
- Move()
- Retrieve()
- Search()
- Describe()
- Audit()

Depending on the requirements of the client these methods may be used together to provide required functionality. For example, to select from a list of Agent Team resources and then display the list of Agents within the selected Agent Team, you may:

1. Retrieve - Items of type Agent Team
2. <await user selection>
### Overview

3. Retrieve - Agent Members of the selected Agent Team

#### About Enterprise-Level Caching

Due to the distributed environment in which the Unified CCMP is deployed, some APIs rely on an enterprise level caching mechanism to provide enhanced scalability and response times. This behavior may mean that when polling services in quick succession data may take some time to update across all servers in the cluster and between the result sets of each API.

Table 3-1 shows which APIs provide direct database level access and which APIs rely on cached data from the enterprise level cache.

#### Table 3-1  Direct and Cached APIs

<table>
<thead>
<tr>
<th>API</th>
<th>Direct Database Access</th>
<th>Enterprise Level Cache</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Update()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Delete()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Save()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Move()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Retrieve()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Search()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Describe()</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Audit()</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

Notifications sent from the Unified CCMP application server also rely on updates from the enterprise level cache so will only be sent when this cache is updated.

#### Specification

**SOAP**

The WSDL file for the Resource Management Web Service is located on the Web/Application server here:

```
```

where `<server>` is the name of the Web/Application server.
REST

The REST protocol for each type and API is given in the section describing the type or API.

Restrictions

Array Limits

The Unified CCMP Resource Management Web Service APIs impose limits on the size of the array parameters that can be passed to the API methods to protect the server and the clients. These limits are:

<table>
<thead>
<tr>
<th>Method</th>
<th>Input Limit</th>
<th>Output Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit()</td>
<td>255 Resource Keys</td>
<td>255 Resource Audits</td>
</tr>
<tr>
<td>Create()</td>
<td>100 Resources or Resource Members</td>
<td>255 Request Results</td>
</tr>
<tr>
<td>Update()</td>
<td>100 Resources or Resource Members</td>
<td>255 Request Results</td>
</tr>
<tr>
<td>Delete()</td>
<td>255 Resource Keys</td>
<td>255 Request Results</td>
</tr>
<tr>
<td>Save()</td>
<td>100 Resources or Resource Members</td>
<td>255 Request Results</td>
</tr>
<tr>
<td>Move()</td>
<td>100 Resource Keys</td>
<td>255 Request Results</td>
</tr>
<tr>
<td>Describe()</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Retrieve()</td>
<td>255 Resource Keys</td>
<td>255 Resources</td>
</tr>
<tr>
<td>Search()</td>
<td>None</td>
<td>255 Resources</td>
</tr>
</tbody>
</table>

REST URI Segment Limit

Each segment in a REST URI is limited to a maximum of 255 characters (including query parameters). If you exceed this limit, then the Web Service stack will return an HTML error page containing an Endpoint not Found error.
## Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Exception</td>
<td>General exception; represents an Operating System fault, for example, a Microsoft WCF web service internal error.</td>
</tr>
<tr>
<td>2</td>
<td>ApplicationException</td>
<td>General catch all application error; typically represents an unspecified fault in third party libraries if not explicitly trapped.</td>
</tr>
<tr>
<td>40</td>
<td>ArgumentException</td>
<td>Error(s) detected in the parameters passed in the web service request.</td>
</tr>
<tr>
<td>50</td>
<td>SerializationException</td>
<td>Error(s) detected in the data structure passed in the web service request. This can either be missing or erroneous fields or a data type conversion in the contents of a field(s).</td>
</tr>
<tr>
<td>100</td>
<td>ConcurrencyConflictException</td>
<td>The resource to be updated has already been updated by another process, that is, its ChangeStamp field is too old.</td>
</tr>
<tr>
<td>101</td>
<td>ServiceNotReadyException</td>
<td>The system service(s) are not yet ready to accept web service requests.</td>
</tr>
<tr>
<td>500</td>
<td>InvalidKeywordException</td>
<td>Search request error: the system does not recognize the supplied search keyword term, for example, specifying type rather than type.</td>
</tr>
<tr>
<td>501</td>
<td>InvalidConditionException</td>
<td>Search request error: the system could not parse the supplied search term condition, for example, specifying type: Agent rather than type: Agent.</td>
</tr>
<tr>
<td>502</td>
<td>FieldConversionException</td>
<td>System could not parse the contents of a field. This is an application level error related to the system SerializationException and means that although the data type is correct, its value is incorrect in that context. For example, in: Agent.PeripheralName: 12~@FG 12~@FG is a valid string but an invalid value for this field type.</td>
</tr>
</tbody>
</table>

---

Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Exception</td>
<td>General exception; represents an Operating System fault, for example, a Microsoft WCF web service internal error.</td>
</tr>
<tr>
<td>2</td>
<td>ApplicationException</td>
<td>General catch all application error; typically represents an unspecified fault in third party libraries if not explicitly trapped.</td>
</tr>
<tr>
<td>40</td>
<td>ArgumentException</td>
<td>Error(s) detected in the parameters passed in the web service request.</td>
</tr>
<tr>
<td>50</td>
<td>SerializationException</td>
<td>Error(s) detected in the data structure passed in the web service request. This can either be missing or erroneous fields or a data type conversion in the contents of a field(s).</td>
</tr>
<tr>
<td>100</td>
<td>ConcurrencyConflictException</td>
<td>The resource to be updated has already been updated by another process, that is, its ChangeStamp field is too old.</td>
</tr>
<tr>
<td>101</td>
<td>ServiceNotReadyException</td>
<td>The system service(s) are not yet ready to accept web service requests.</td>
</tr>
<tr>
<td>500</td>
<td>InvalidKeywordException</td>
<td>Search request error: the system does not recognize the supplied search keyword term, for example, specifying type rather than type.</td>
</tr>
<tr>
<td>501</td>
<td>InvalidConditionException</td>
<td>Search request error: the system could not parse the supplied search term condition, for example, specifying type: Agent rather than type: Agent.</td>
</tr>
<tr>
<td>502</td>
<td>FieldConversionException</td>
<td>System could not parse the contents of a field. This is an application level error related to the system SerializationException and means that although the data type is correct, its value is incorrect in that context. For example, in: Agent.PeripheralName: 12~@FG 12~@FG is a valid string but an invalid value for this field type.</td>
</tr>
</tbody>
</table>

---

Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Exception</td>
<td>General exception; represents an Operating System fault, for example, a Microsoft WCF web service internal error.</td>
</tr>
<tr>
<td>2</td>
<td>ApplicationException</td>
<td>General catch all application error; typically represents an unspecified fault in third party libraries if not explicitly trapped.</td>
</tr>
<tr>
<td>40</td>
<td>ArgumentException</td>
<td>Error(s) detected in the parameters passed in the web service request.</td>
</tr>
<tr>
<td>50</td>
<td>SerializationException</td>
<td>Error(s) detected in the data structure passed in the web service request. This can either be missing or erroneous fields or a data type conversion in the contents of a field(s).</td>
</tr>
<tr>
<td>100</td>
<td>ConcurrencyConflictException</td>
<td>The resource to be updated has already been updated by another process, that is, its ChangeStamp field is too old.</td>
</tr>
<tr>
<td>101</td>
<td>ServiceNotReadyException</td>
<td>The system service(s) are not yet ready to accept web service requests.</td>
</tr>
<tr>
<td>500</td>
<td>InvalidKeywordException</td>
<td>Search request error: the system does not recognize the supplied search keyword term, for example, specifying type rather than type.</td>
</tr>
<tr>
<td>501</td>
<td>InvalidConditionException</td>
<td>Search request error: the system could not parse the supplied search term condition, for example, specifying type: Agent rather than type: Agent.</td>
</tr>
<tr>
<td>502</td>
<td>FieldConversionException</td>
<td>System could not parse the contents of a field. This is an application level error related to the system SerializationException and means that although the data type is correct, its value is incorrect in that context. For example, in: Agent.PeripheralName: 12~@FG 12~@FG is a valid string but an invalid value for this field type.</td>
</tr>
</tbody>
</table>

---

Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Exception</td>
<td>General exception; represents an Operating System fault, for example, a Microsoft WCF web service internal error.</td>
</tr>
<tr>
<td>2</td>
<td>ApplicationException</td>
<td>General catch all application error; typically represents an unspecified fault in third party libraries if not explicitly trapped.</td>
</tr>
<tr>
<td>40</td>
<td>ArgumentException</td>
<td>Error(s) detected in the parameters passed in the web service request.</td>
</tr>
<tr>
<td>50</td>
<td>SerializationException</td>
<td>Error(s) detected in the data structure passed in the web service request. This can either be missing or erroneous fields or a data type conversion in the contents of a field(s).</td>
</tr>
<tr>
<td>100</td>
<td>ConcurrencyConflictException</td>
<td>The resource to be updated has already been updated by another process, that is, its ChangeStamp field is too old.</td>
</tr>
<tr>
<td>101</td>
<td>ServiceNotReadyException</td>
<td>The system service(s) are not yet ready to accept web service requests.</td>
</tr>
<tr>
<td>500</td>
<td>InvalidKeywordException</td>
<td>Search request error: the system does not recognize the supplied search keyword term, for example, specifying type rather than type.</td>
</tr>
<tr>
<td>501</td>
<td>InvalidConditionException</td>
<td>Search request error: the system could not parse the supplied search term condition, for example, specifying type: Agent rather than type: Agent.</td>
</tr>
<tr>
<td>502</td>
<td>FieldConversionException</td>
<td>System could not parse the contents of a field. This is an application level error related to the system SerializationException and means that although the data type is correct, its value is incorrect in that context. For example, in: Agent.PeripheralName: 12~@FG 12~@FG is a valid string but an invalid value for this field type.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Identifier</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Various</td>
<td>EntityValidationException</td>
<td>The system has detected an error in the information supplied in the web service call. This is not an OS or field type error but an error in the composite data making up the whole request.</td>
</tr>
<tr>
<td>Various</td>
<td>ExonyApplicationException</td>
<td>Last try application exception.</td>
</tr>
<tr>
<td>50150</td>
<td>NoMembershipEffectivenessOverlap</td>
<td>A type of EntityValidationException error. Indicates that there is an error in the supplied member effective date-time range where the effective_from is newer than the effective_to.</td>
</tr>
<tr>
<td>50159</td>
<td>RequiredFieldNotSpecified</td>
<td>A type of EntityValidationException error. Indicates that the contents of a field have been incorrectly sent as empty. For example, an IPEndpoint.MACAddress field has been supplied as an empty string.</td>
</tr>
<tr>
<td>50160</td>
<td>RegularExpressionValidationFailed</td>
<td>A type of EntityValidationException error. Indicates that the contents of a field have failed a check against the regular expression mask used to validate its contents. For example, an IPEndpoint.MACAddress field does not match the mask needed for MAC addresses.</td>
</tr>
<tr>
<td>50161</td>
<td>FieldValueOutOfRange</td>
<td>A type of EntityValidationException error. Indicates that the contents of a field are outside the boundary limits.</td>
</tr>
<tr>
<td>100000</td>
<td>MissingMember</td>
<td>Future use. Indicates that a mandatory member was not supplied in the web service.</td>
</tr>
<tr>
<td>100001</td>
<td>AgentMultiplePersonality</td>
<td>Future use.</td>
</tr>
<tr>
<td>100002</td>
<td>PeripheralRequired</td>
<td>Future use.</td>
</tr>
<tr>
<td>100003</td>
<td>RoutingClientRequired</td>
<td>Future use.</td>
</tr>
<tr>
<td>100004</td>
<td>IPEndpointMultipleLines</td>
<td>A type of EntityValidationException error. Indicates that an attempt has been made to provision more than one line on a device.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Identifier</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>100005</td>
<td>UnknownClusterResource</td>
<td>A type of <code>ExonyApplicationException</code> error. Indicates that the supplied cluster resource type is not a supported type.</td>
</tr>
<tr>
<td>100006</td>
<td>AgentLimitTeamCapacityExceeded</td>
<td>Future use.</td>
</tr>
<tr>
<td>100007</td>
<td>SkillgroupPTA</td>
<td>A type of <code>EntityValidationException</code> error. Indicates that for voice, skill groups must let the system pick the agent.</td>
</tr>
<tr>
<td>100039</td>
<td>CannotFindDefaultRoutingClient</td>
<td>A type of <code>EntityValidationException</code> error. Indicates that the tenant referred to by the supplied folder id does not have access to the correct type of routing client needed for this type of request.</td>
</tr>
<tr>
<td>100040</td>
<td>CannotFindDefaultPeripheral</td>
<td>A type of <code>EntityValidationException</code> error. Indicates that the tenant referred to by the supplied folder id does not have access to the correct type of peripheral needed for this type of request.</td>
</tr>
<tr>
<td>100041</td>
<td>CannotFindDefaultOutboundRoutingClient</td>
<td>A type of <code>EntityValidationException</code> error. Indicates that the tenant referred to by the supplied folder id does not have access to the correct type of outbound routing client needed for this type of request.</td>
</tr>
<tr>
<td>100050</td>
<td>CannotUpdateMaxAttempts</td>
<td>A type of <code>EntityValidationException</code> error. Indicates that the system has rejected an attempt to update a campaign when it is running.</td>
</tr>
<tr>
<td>100054</td>
<td>TooLongSkillGroupName</td>
<td>A type of <code>EntityValidationException</code> error. A special case of <code>Internal Name</code> validation failure, where the associated Peripheral has a sub-skill group configured in which case the usual limit is reduced to 28 characters.</td>
</tr>
<tr>
<td>100058</td>
<td>LoginNameChangeNotSupported</td>
<td>A type of <code>EntityValidationException</code> error. Indicates that the supplied equipment login name cannot be provisioned at this time, typically due to configuration on the equipment.</td>
</tr>
</tbody>
</table>
### Error Codes

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>100074</td>
<td>NoEquipmentMappedToTenant</td>
<td>A type of EntityValidationException error. Indicates that the system has not found a cluster resource or resources of the correct type to match the supplied web request. For example, a create call type request has been supplied but a Unified CCE matching that request has not been found. This can happen if the cluster type instance has not been added by the ICE tool (See Installation and Configuration Guide for Cisco Unified Contact Center Management Portal) or the user does not have the correct security permissions to get access to it.</td>
</tr>
<tr>
<td>100097</td>
<td>BadRequestException</td>
<td>The requested action will cause a Precision Queue to have no Precision Queue Steps. A Precision Queue must have at least one Precision Queue Step.</td>
</tr>
<tr>
<td>250000</td>
<td>EnterpriseNameAlreadyExistsException</td>
<td>The system has detected that a resource.InternalName for this resource type and equipment cluster already exists. This is an alternative primary key that must be unique.</td>
</tr>
<tr>
<td>250001</td>
<td>LoginNameAlreadyExistsException</td>
<td>The system has detected that a User.LoginName already exists. The LoginName must be globally unique across the installation.</td>
</tr>
</tbody>
</table>

### Data Types

This section lists common data types used in the Resource Management Web Service APIs. The data types in this section provide type-safe constructs for standard entities.

### Resource Hierarchy Description

The item and member types are represented by the resource class hierarchy shown in Figure 3-1.
The top level abstract class, **Entry**, provides a simple-type definition for an entity. Requests to retrieve contact center objects such as Agents and Skill Groups take the form of these common objects.

An entity can be:
- an **Item**
- a **Member**, which represents a membership between two items, and is one of:
  - a physical many-to-many relationship: for example, an Agent may be skilled in one or more Skill Groups
  - a type-2 SCD relationship, for example, an Agent may be part of an Agent Team during May 2010 and be part of a different Agent Team from June 2010 to Feb 2011.

Type-2 SCD relationships are rarely used in provisioning and should usually be filtered from search queries using the “latest:1” search term.

Table 3-2 defines the classes in the resource class hierarchy above.
Table 3-2  Resource Class Hierarchy Definitions

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry</td>
<td>The Entry class represents a primary Unified CCMP entity or membership.</td>
</tr>
<tr>
<td>Item</td>
<td>An entity that can be created in Unified CCMP or imported into Unified CCMP.</td>
</tr>
<tr>
<td>Member</td>
<td>The membership between two entities, either physical or type-2 temporal.</td>
</tr>
<tr>
<td>System</td>
<td>A primary entity that is native to Unified CCMP only with no representation on other systems (a system resource). Examples include Folder, Group and User entities.</td>
</tr>
<tr>
<td>SystemMember</td>
<td>The membership between two system entities, for example a UserFolder member.</td>
</tr>
<tr>
<td>DimensionItem</td>
<td>A primary entity that has one or more representations on other systems (a remote resource). Examples include Agent, AgentTeam and DialedNumber entities.</td>
</tr>
<tr>
<td>DimensionMember</td>
<td>The membership between two dimension (or remote resource) entities. Examples include AgentPersonMember and DialedNumberCallTypeMember.</td>
</tr>
</tbody>
</table>

Common Fields and Classes

This section defines some common fields and classes used by several classes.

Member Types

This section describes the items that define membership between entities.

<table>
<thead>
<tr>
<th>Member Type</th>
<th>Internal Name</th>
<th>REST Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Agent Desktop Member</td>
<td>MT_AGENT_AGENT_DESKTOP_MEMBER</td>
<td>agent-agent-desktop-member</td>
</tr>
<tr>
<td>Agent Agent Team Member</td>
<td>MT_AGENT_AGENT_TEAM_MEMBER</td>
<td>agent-agent-team-member</td>
</tr>
<tr>
<td>Agent Desktop Dialed Number Member</td>
<td>MT_AGENT_DESKTOP_DIALED_NUMBER_MEMBER</td>
<td>agent-desktop-dialed-number-member</td>
</tr>
<tr>
<td>Agent Peripheral Member</td>
<td>MT_AGENT_PERIPHERAL_MEMBER</td>
<td>agent-peripheral-member</td>
</tr>
<tr>
<td>Member Type</td>
<td>Internal Name</td>
<td>REST Parameter</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>Agent Person Member</td>
<td>MT_AGENT_PERSON_MEMBER</td>
<td>agent-person-member</td>
</tr>
<tr>
<td>Agent Precision Attribute Member</td>
<td>MT_AGENT_PRECISION_ATTRIBUTE_MEMBER</td>
<td>agent-precision-attribute-member</td>
</tr>
<tr>
<td>Agent Skill Group Member</td>
<td>MT_AGENT_SKILLGROUP_MEMBER</td>
<td>agent-skill-group-member</td>
</tr>
<tr>
<td>Agent Team Dialed Number Member</td>
<td>MT_AGENT_TEAM_DIALED_NUMBER_MEMBER</td>
<td>agent-dialed-number-member</td>
</tr>
<tr>
<td>Call Type Routing Script Member</td>
<td>MT_CALL_TYPE_ROUTING_SCRIPT_MEMBER</td>
<td>call-type-routing-script-member</td>
</tr>
<tr>
<td>Device Profile Directory Number Member</td>
<td>MT_DEVICE_PROFILE_DIRECTORY_NUMBER_MEMBER</td>
<td>device-profile-directory-number-member</td>
</tr>
<tr>
<td>Device Profile IP Endpoint Member</td>
<td>MT_DEVICE_PROFILE_IP_ENDPOINT_MEMBER</td>
<td>device-profile-ip-endpoint-member</td>
</tr>
<tr>
<td>Device Profile Person Member</td>
<td>MT_DEVICE_PROFILE_PERSON_MEMBER</td>
<td>device-profile-person-member</td>
</tr>
<tr>
<td>Dialed Number Call Type Member</td>
<td>MT_DIALED_NUMBER_CALL_TYPE_MEMBER</td>
<td>dialed-number-call-type-member</td>
</tr>
<tr>
<td>Dialed Number Routing Client Member</td>
<td>MT_DIALED_NUMBER_ROUTING_CLIENT_MEMBER</td>
<td>dialed-number-routing-client-member</td>
</tr>
<tr>
<td>Dialed Number Media Routing Domain Member</td>
<td>MT_DIALED_NUMBER_MEDIA_ROUTING_DOMAIN_MEMBER</td>
<td>dialed-number-media-routing-domain-member</td>
</tr>
<tr>
<td>Group Group Member</td>
<td>MT_GROUP_GROUP_MEMBER</td>
<td>group-group-member</td>
</tr>
<tr>
<td>Member Type</td>
<td>Internal Name</td>
<td>REST Parameter</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>------------------------------------------------------------</td>
<td>--------------------------------------------------------</td>
</tr>
<tr>
<td>IP Endpoint Button Template Member</td>
<td>MT_IP_ENDPOINT_BUTTON_TEMPLATE_MEMBER</td>
<td>i p-endpoint-button-template-member</td>
</tr>
<tr>
<td>IP Endpoint Calling Search Space Member</td>
<td>MT_IP_ENDPOINT_CALLING_SEARCH_SPACE_MEMBER</td>
<td>i p-endpoint-calling-search-space-member</td>
</tr>
<tr>
<td>IP Endpoint Device Pool Member</td>
<td>MT_IP_ENDPOINTDEVICE_POOL_MEMBER</td>
<td>i p-endpoint-device-pool-member</td>
</tr>
<tr>
<td>IP Endpoint Directory Number Member</td>
<td>MT_IP_ENDPOINTDIRECTORY_NUMBER_MEMBER</td>
<td>i p-endpoint-directory-number-member</td>
</tr>
<tr>
<td>IP Endpoint Peripheral Member</td>
<td>MT_IP_ENDPOINTPERIPHERAL_MEMBER</td>
<td>i p-endpoint-peripheral-member</td>
</tr>
<tr>
<td>Label Dialed Number Member</td>
<td>MT_LABEL_DIALED_NUMBER_MEMBER</td>
<td>label-dialed-number-member</td>
</tr>
<tr>
<td>Label Routing Client Member</td>
<td>MT_LABEL_ROUTING_CLIENT_MEMBER</td>
<td>label-routing-client-member</td>
</tr>
<tr>
<td>Network Vru Script Network Vru Member</td>
<td>MT_NETWORK_VRU_SCRIPT_NETWORK_VRU_MEMBER</td>
<td>network-vru-script-network-vru-member</td>
</tr>
<tr>
<td>Precision Queue Bucket Interval Member</td>
<td>MT_PRECISION_QUEUE_BUCKET_INTERVAL_MEMBER</td>
<td>precision-queue-bucket-interval-member</td>
</tr>
<tr>
<td>Precision Queue Step Precision Attribute Member</td>
<td>MT_PRECISION_QUEUE_STEP_PRECISION_ATTRIBUTE_MEMBER</td>
<td>precision-queue-step-precision-attribute-member</td>
</tr>
<tr>
<td>Precision Queue Step Precision Queue Member</td>
<td>MT_PRECISION_QUEUE_STEP_PRECISION_QUEUE_MEMBER</td>
<td>precision-queue-step-precision-queue-member</td>
</tr>
<tr>
<td>Member Type</td>
<td>Internal Name</td>
<td>REST Parameter</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------------------------------------</td>
<td>-----------------------------------------</td>
</tr>
<tr>
<td>Query Rule Campaign Member</td>
<td>MTQUERY_RULE_CAMPAIGN_MEMBER</td>
<td>query-rule-campaign-member</td>
</tr>
<tr>
<td>Route Partition Calling Search Space Member</td>
<td>MTROUTEPARTITIONCALLINGSSEARCHSPACE_MEMBER</td>
<td>route-partition-calling-search-space-member</td>
</tr>
<tr>
<td>Route Skill Group Member</td>
<td>MT_ROUTESKILLGROUP_MEMBER</td>
<td>route-skill-group-member</td>
</tr>
<tr>
<td>Service Enterprise Service Member</td>
<td>MTSERVICESERVICE_MEMBER</td>
<td>service-enterprise-service-member</td>
</tr>
<tr>
<td>Service Media Routing Domain Member</td>
<td>MTSERVICEMEDIALROUTINGDOMAIN_MEMBER</td>
<td>service-media-routing-domain-member</td>
</tr>
<tr>
<td>Service Peripheral Member</td>
<td>MT_SERVICEPERIPHERAL_MEMBER</td>
<td>service-peripheral-member</td>
</tr>
<tr>
<td>Skill Group Campaign Member</td>
<td>MTSKILLGROUP_CAMPAIGN_MEMBER</td>
<td>skill-group-campaign-member</td>
</tr>
<tr>
<td>Skill Group Enterprise Skill Group Member</td>
<td>MTSKILLGROUP_ENTERPRISE_SKILLGROUP_MEMBER</td>
<td>skill-group-enterprise-skill-group-member</td>
</tr>
<tr>
<td>Skill Group Media Routing Domain Member</td>
<td>MTSKILLGROUP_MEDIA_ROUTING_DOMAIN_MEMBER</td>
<td>skill-group-media-routing-domain-member</td>
</tr>
<tr>
<td>Skill Group Peripheral Member</td>
<td>MTSKILLGROUPPERIPHERAL_MEMBER</td>
<td>skill-group-peripheral-member</td>
</tr>
<tr>
<td>Skill Group Service Member</td>
<td>MTSKILLGROUPSERVICE_MEMBER</td>
<td>skill-group-service-member</td>
</tr>
<tr>
<td>User Group Member</td>
<td>MTUSERGROUP_MEMBER</td>
<td>user-group-member</td>
</tr>
</tbody>
</table>
Name-Value Pair

The `NameValuePair` class is used to provide type-safe collections of information to be communicated to and from the server.

In the class descriptions in the rest of this chapter, items that are encoded as name-value pairs rather than named elements are denoted by (n-v) after the data type.

**Properties**

The exposed properties on the `NameValuePair` class are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>String</td>
<td>The name field for the name-value pair.</td>
<td>Yes</td>
</tr>
<tr>
<td>Value</td>
<td>String</td>
<td>The value field for the name-value pair.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| Storage Type | `NameValuePair` StorageType | The type of storage that this name-value pair uses. One of:
- **Standard**. Used by Unified CCMP
- **Custom**. Used by the client application to store related information, for example, a customer-specific employment number may be stored with an Agent resource. | No         |

The order of the elements in the `NameValuePair` class is significant. The elements must be supplied to the Web Service APIs in the order above, for both SOAP and REST protocols.

**Status**

The `status` field is used to report the status of a remote resource. Figure 3-2 shows the state transitions for a remote resource.
Figure 3-2  Resource State Transitions

The valid status values are:

<table>
<thead>
<tr>
<th>Status</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ready</td>
<td>R</td>
<td>The item is in ready state, the state of the object to which this status relates is the same as the state on the remote system for example, on Unified CCE.</td>
</tr>
<tr>
<td>Pending</td>
<td>S</td>
<td>The Pending (or Synchronizing) status means that a change has been made that has not yet been reflected on the remote system. Newly created items will be put into this state initially, before progressing to the Ready state when they have successfully been created on the remote system. for example, on Unified CCE or Unified CM.</td>
</tr>
<tr>
<td>Pending Delete</td>
<td>P</td>
<td>This status represents that the item has been deleted from Unified CCMP but has not yet been removed on the remote system. Once the delete operation is complete then the status will change to Deleted meaning that the resource has been removed from the remote system (for example, from Unified CCE).</td>
</tr>
<tr>
<td>Deleted</td>
<td>D</td>
<td>This item has been deleted on the remote system.</td>
</tr>
<tr>
<td>Error</td>
<td>E</td>
<td>An error has occurred whilst trying to change the item on the remote system. The current state in the Unified CCMP database does not reflect the state on the remote system. Errors should be corrected and the resource should be re-submitted.</td>
</tr>
</tbody>
</table>

Resource Hierarchy Classes

This section describes the classes that make up the resource hierarchy.
**Entry**

The `Entry` class represents a primary Unified CCMP entity or membership.

**Properties**

The exposed properties on the `Entry` class are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity</td>
<td>String</td>
<td>The identity of the related entity. May be an integer value for dimension entities (as represented by the <code>ITEM_URN</code> field from <code>TB_DIM_ITEM</code> or <code>MEMBER_URN</code> field from <code>TB_DIM_MEMBER</code>) or a GUID value for system entities (for example, <code>USER_ID</code> from <code>TB_SEC_USER</code>).</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| Type         | String    | A string identifying the item type. One of the following:  
• a remote resource type  
• a system resource type  
• a member type  
• a pkey type  
• a pkey member type. | Yes |
| EffectiveFrom| DateTime  | The UTC date that the entity will become effective or has become effective in the contact center environment. | Yes |
| EffectiveTo  | DateTime  | The UTC date that the entity will expire or has expired and been removed or deactivated in the contact center environment. | Yes |
| Status       | String    | The provisioning status of the resource. | Yes |
| Changestamp  | Int       | The concurrency change stamp for this resource. For provisionable resources, the system checks the value of this field to determine whether the provisioning request is allowed, that is, whether another client has updated the resource since it was retrieved or searched for. Not required for `Create()`, but must be supplied for `Update()`. | Yes |
| Fields       | NameValue Pair[] | A collection of the fields that have been set or may be set on the current resource | Yes |
### Element Name | Data Type | Description                                                                                                                                 | Required?
--- | --- | --- | ---
Equipment Mapping | Resource[] | An array containing either ItemPkey or MemberPkey elements, specifying resource details that are specific to an equipment instance. Only required if one of the following is true:
- Type is a remote resource types and the resource is mapped to more than one equipment instance.
- Type is a resource membership type and the membership is mapped to more than one equipment instance. | No

**Note**

The order of the elements in the Entry class is significant. The elements must be supplied to the Web Service APIs in the order above, for both SOAP and REST protocols.

### Item

The Item class represents an entity that can be created in Unified CCMP or imported into Unified CCMP. An Item is always has a name, and a location in the security folder tree.

#### Properties

The exposed properties on the Item class are:

| Element Name | Data Type | Description                                                                 | Required? |
--- | --- | --- | ---
Inherits from: Entry |  |  |  |
FolderId | Guid (n-v pair) | The identity of the related folder. | Yes |
Name | String (n-v pair) | This is the friendly name for this resource, typically used for reporting processes. | Yes |
### Data Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherit from: Entry</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internal Name</td>
<td>String (n-v pair)</td>
<td>The name of the entity on underlying equipment. For example, in Unified CCE this represents the Cisco EnterpriseName field. If the resource exists on multiple equipment instances, this is the identifier of the primary cluster resource (defined as the first cluster on which the item is provisioned).</td>
<td>No</td>
</tr>
<tr>
<td>Description</td>
<td>String (n-v pair)</td>
<td>The description of this instance of the resource.</td>
<td>No</td>
</tr>
<tr>
<td>MappedClusterResources</td>
<td>String (n-v pair)</td>
<td>The identifier of the underlying equipment on which the resource exists. If the resource exists on multiple equipment instances, this is a comma-separated list of the equipment instances. Default: System will select based on Tenant.</td>
<td>No</td>
</tr>
<tr>
<td>MappedClusterResourceDetails</td>
<td>String (n-v pair)</td>
<td>The details of the underlying equipment on which the resource exists. If the resource exists on multiple equipment instances, this is a comma-separated list of the details for each equipment instance.</td>
<td>No</td>
</tr>
</tbody>
</table>

### Dimension Item

The `DimensionItem` class represents a reporting dimension or a provisionable entity. It may be created in Unified CCMP or imported into Unified CCMP.

The exposed properties on the `DimensionItem` class are:
## Data Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: Item</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CurrencyUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. The currency associated with this dimension item. Default: -1.</td>
<td>No</td>
</tr>
<tr>
<td>GeographyUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. The geographical location associated with this dimension item. Default: -1.</td>
<td>No</td>
</tr>
<tr>
<td>HierarchyParentUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. The type-1 Hierarchy for this dimension item. Always -1.</td>
<td>No</td>
</tr>
<tr>
<td>ItemBusinessUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. Type-2 Business id for this dimension item. Default: -1.</td>
<td>No</td>
</tr>
<tr>
<td>PrimaryLanguageUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. The first language in order of importance for this dimension item. Default: -1.</td>
<td>No</td>
</tr>
<tr>
<td>SecondaryLanguageUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. The second language in order of importance for this dimension item. Default: -1.</td>
<td>No</td>
</tr>
<tr>
<td>TertiaryLanguageUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. The third language in order of importance for this dimension item. Default: -1.</td>
<td>No</td>
</tr>
<tr>
<td>TenantBizUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. Item Business Urn of the tenant dimension associated with this dimension.</td>
<td>No</td>
</tr>
<tr>
<td>TimeZoneUrn</td>
<td>Int (n-v pair)</td>
<td>Read-only. The time zone in which the resource is located.</td>
<td>No</td>
</tr>
</tbody>
</table>

### Member

The **Member** class represents a membership between two entities. One of the entities in the membership will be designated the parent entity, and the other the child. Often,
the relationship will be many-many, but typically, the parent member will have many more children than the child member has parents. For example, in a Skill Group to Agent relationship, the Skill Group is the parent and the Agent the child. Skill Group can contain a number of Agents, and an Agent can belong to a number of Skill Groups, but typically, a Skill Group will contain many Agents, whereas an Agent will belong to a just a few Skill Groups.

**Properties**

The exposed properties of the **Member** class are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits from: Entry</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ParentId</td>
<td>Int (n-v pair)</td>
<td>This is the id of the resource which is the parent resource for the membership.</td>
<td>Yes</td>
</tr>
<tr>
<td>ChildId</td>
<td>Int (n-v pair)</td>
<td>This is the id of the resource which is the child resource for the membership.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Dimension Member**

The **DimensionMember** class inherits from the **Member** class and represents a membership between **DimensionItem** (or remote resource) entities. A key characteristic is that these memberships normally have type-2 effective dating enabled.

**Properties**

The exposed properties on the **DimensionMember** class are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: Member</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LatestParentItemUrn</td>
<td>Int (n-v pair)</td>
<td>This is the id of the resource which is the most recent parent resource for the membership.</td>
<td>Yes</td>
</tr>
<tr>
<td>LatestChildItemUrn</td>
<td>Int (n-v pair)</td>
<td>This is the id of the resource which is the most recent child resource for the membership.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**System Member**

The **SystemMember** class inherits from the **Member** class and represents the membership between system resource entities.

The exposed properties on the **SystemMember** class are:
### Other Common Classes

This section describes other common classes used by the Resource Management Web Service APIs.

#### Resource Meta

The `ResourceMeta` class describes a field for an entity of a given type.

**Properties**

The exposed properties on the `ResourceMeta` class are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>String</td>
<td>The name of the field that may be set/read on the entity</td>
</tr>
<tr>
<td>DataType</td>
<td>String</td>
<td>The data type of the field that may be set/read on the entity</td>
</tr>
<tr>
<td>Length</td>
<td>Int</td>
<td>The length of the data type.</td>
</tr>
<tr>
<td>Required</td>
<td>Bool</td>
<td>States if the field described is required or optional</td>
</tr>
<tr>
<td>DefaultValue</td>
<td>String</td>
<td>The default value as a string for the field that has been specified</td>
</tr>
<tr>
<td>CanCreate</td>
<td>Bool</td>
<td>Determines whether this field is required or valid for a create operation.</td>
</tr>
<tr>
<td>CanEdit</td>
<td>Bool</td>
<td>Determines if the field can be changed as part of an update operation.</td>
</tr>
<tr>
<td>FieldGrouping</td>
<td>String</td>
<td>Indicates how this field can be grouped for client UIs or workflows. One of:</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Common.</strong> The field is common to all resource types</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Specific.</strong> The field is specific to this resource type.</td>
</tr>
</tbody>
</table>
### Data Types

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| Field dType | String | Indicates the usage of this field for automated workflow clients. One of:  
- **Key**: Field contains an Id or Urn to another resource instance.  
- **Field**: The field is an attribute of this type of resource.  
- **Virtual**: The field is defined by a calculation or algorithm.  
- **Xml**: The field contains an XML data block specific to this resource type.  
- **Equipment**: The field contains equipment cluster reference. |
| HelpText | String | A description of the use of this field. |

#### Resource Field Meta

The `ResourceFieldMeta` class describes the associated fields for an entity of a given type.

**Properties**

The exposed properties on the `ResourceFieldMeta` class are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| Type     | String   | The resource type to which this metadata description applies. One of the following resource types:  
- a remote resource type  
- a system resource type  
- a member type  
- a pkey map type  
- a pkey map member type. |
| CanCreate | Bool     | An instance of this resource may be created in the Unified CCMP database. |
| CanEdit   | Bool     | An instance of this resource may be edited. |
| CanDelete | Bool     | An instance of this resource may be deleted from the Unified CCMP database. |
Chapter 3: Resource Management Web Service

### Data Types

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanProvision</td>
<td>Bool</td>
<td>An instance of this resource will be provisioned into one or more equipment native objects on the underlying CC equipment.</td>
</tr>
<tr>
<td>IsHidden</td>
<td>Bool</td>
<td>An instance of this class may be hidden, that is, reserved for internal use by Unified CCMP only</td>
</tr>
<tr>
<td>IsSystem</td>
<td>Bool</td>
<td>An instance of this class may be manipulated only by user accounts of Host Administrator only</td>
</tr>
<tr>
<td>ClusterResourceTypes</td>
<td>String</td>
<td>A comma separated string of the underlying equipment types supported by this resource.</td>
</tr>
<tr>
<td>Fields</td>
<td>ResourceMeta[]</td>
<td>The description of the fields that make up this entity type.</td>
</tr>
</tbody>
</table>

### Resource Key

The `ResourceKey` class represents the entity identity when performing a delete, retrieve, search or audit operation.

**Properties**

The exposed properties on the `ResourceKey` class are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity</td>
<td>String</td>
<td>The identity of the related item. May be an integer value for dimension entities or a GUID value for system entities.</td>
</tr>
<tr>
<td>ResourceType</td>
<td>String</td>
<td>The item type as defined in the item types listed out later in this document</td>
</tr>
</tbody>
</table>

### Resource Audit

The `ResourceAudit` class describes an audit record that has been logged as part of a provisioning request.

**Properties**

The exposed properties on the `ResourceAudit` class are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuditType</td>
<td>String</td>
<td>The type of audit (one of “System” or “Resource”).</td>
</tr>
<tr>
<td>EventDateTime</td>
<td>String</td>
<td>The UTC Date Time at which the audited event was entered into the system.</td>
</tr>
</tbody>
</table>
### Data Types

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserName</td>
<td>String</td>
<td>The name of the user that caused the creation of the audit. May be System if the audit operation was performed by a Unified CCMP scheduled process such as Data Importer.</td>
</tr>
<tr>
<td>Description</td>
<td>String</td>
<td>The text indicating which provisioning operation took place.</td>
</tr>
<tr>
<td>ResourceIdentity</td>
<td>String</td>
<td>The identifier of the resource that is the subject of the audit. Together with the ResourceType is the unique composite identifier of the resource.</td>
</tr>
<tr>
<td>ResourceType</td>
<td>String</td>
<td>The type of the resource that is the subject of the audit. Together with the ResourceIdentity, this is the unique composite identifier of the resource.</td>
</tr>
<tr>
<td>ResourceName</td>
<td>String</td>
<td>The name of the resource that is the subject of the audit.</td>
</tr>
</tbody>
</table>

**Detail fields:**

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserAgentName</td>
<td>String</td>
<td>The name of the user agent that invoked the request that caused the audited provisioning request.</td>
</tr>
<tr>
<td>ClusterResource</td>
<td>String</td>
<td>The remote cluster(s) on which the provisioning operation occurred.</td>
</tr>
<tr>
<td>ChangedFields</td>
<td>XElement</td>
<td>The fields that were modified during the provisioning request.</td>
</tr>
<tr>
<td>AdditionalData</td>
<td>XElement</td>
<td>Any additional supporting data for this audited event.</td>
</tr>
</tbody>
</table>

### Request Result

The RequestResult class holds the status and identity information for a given change. This object contains the identity of any item that was created, deleted or updated, the new status of the item and a collection of error messages if exceptions occurred when the item was changed.

#### Properties

The exposed properties on the RequestResult class are:
### Data Types

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity</td>
<td>String</td>
<td>The identity of the resource that has been changed.</td>
</tr>
<tr>
<td>Type</td>
<td>String</td>
<td>The item type as defined in the item types listed out later in this document.</td>
</tr>
<tr>
<td>Name</td>
<td>String</td>
<td>The name of the resource. Only valid if Type is not a member type.</td>
</tr>
<tr>
<td>Status</td>
<td>String</td>
<td>The status of the resource that has been changed. See section Status.</td>
</tr>
</tbody>
</table>

#### Resource Fault Detail

The `ResourceFaultDetail` class is returned when any method fails and represents the details logged by an exception or other error detected by the server.

**Properties**

The exposed properties on the `ResourceFaultDetail` class are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message</td>
<td>String</td>
<td>A message describing the error that occurred.</td>
</tr>
<tr>
<td>ErrorType</td>
<td>String</td>
<td>The error type.</td>
</tr>
<tr>
<td>ErrorCode</td>
<td>Int</td>
<td>One of the error codes listed in Errors on page 18.</td>
</tr>
</tbody>
</table>

#### Additional Data

The `AdditionalData` class is used by some APIs to send additional name-value fields.

**Properties**

The exposed properties on the `AdditionalData` class are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields</td>
<td>NameValuePair</td>
<td>An array of name-value fields.</td>
</tr>
</tbody>
</table>

#### Page Info

The `PageInfo` class is used by the `Audit()` API to hold paging information about the audit results being returned.
Properties

The exposed properties on the PageInfo class are:

Table 3-3 Page Info Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResultsPerPage</td>
<td>Int</td>
<td>The number of results to be returned in each page of results.</td>
</tr>
<tr>
<td>StartIndex</td>
<td>Int</td>
<td>The index of the first element to be returned.</td>
</tr>
<tr>
<td>TotalResults</td>
<td>Int</td>
<td>The total number of audit results available to be returned.</td>
</tr>
</tbody>
</table>

Resource Audit Results

The ResourceAuditResults class is used by the Audit() API to return audit results.

Properties

The exposed properties on the ResourceAuditResults class are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PageInfo</td>
<td>PageInfo</td>
<td>Information about the collection of audit results being returned.</td>
</tr>
<tr>
<td>Audits</td>
<td>ResourceAudit[]</td>
<td>A collection of ResourceAudit objects containing audit data for the resource id passed in.</td>
</tr>
</tbody>
</table>
Pkey Map Classes

Resource Hierarchy Including Pkey Maps

Provisionable Remote Resource Pkey Map Types

These remote resource pkey map types are fully supported by the Resource Management Web Services.

<table>
<thead>
<tr>
<th>Resource Pkey Map Type</th>
<th>Internal Name</th>
<th>REST Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Pkey</td>
<td>PK_AGENT_PKEY</td>
<td>agent - pkey</td>
</tr>
<tr>
<td>Agent Desktop Pkey</td>
<td>PK_AGENT_DESKTOP_PKEY</td>
<td>agent - desktop-pkey</td>
</tr>
<tr>
<td>Agent Team Pkey</td>
<td>PK_AGENT_TEAM_PKEY</td>
<td>agent - team pkey</td>
</tr>
<tr>
<td>Call Type Pkey</td>
<td>PK_CALL_TYPE_PKEY</td>
<td>call - type-pkey</td>
</tr>
<tr>
<td>Resource Pkey Map Type</td>
<td>Internal Name</td>
<td>REST Parameter</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>----------------------------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>Calling Search Space Pkey</td>
<td>PK_CALLING_SEARCH_SPACE_PKEY</td>
<td>calling-search-space-pkey</td>
</tr>
<tr>
<td>Department Pkey</td>
<td>PK_DEPARTMENT_PKEY</td>
<td>department-pkey</td>
</tr>
<tr>
<td>Device Profile Pkey</td>
<td>PK_DEVICE_PROFILE_PKEY</td>
<td>device-profile-pkey</td>
</tr>
<tr>
<td>Dialed Number Pkey</td>
<td>PK_DIALED_NUMBER_PKEY</td>
<td>dialed-number-pkey</td>
</tr>
<tr>
<td>Directory Number Pkey</td>
<td>PK_DIRECTORY_NUMBER_PKEY</td>
<td>directory-number-pkey</td>
</tr>
<tr>
<td>Enterprise Skill Group Pkey</td>
<td>PK_ENTERPRISE_SKILLGROUP_PKEY</td>
<td>enterprise-skill-group-pkey</td>
</tr>
<tr>
<td>Expanded Call Variable Pkey</td>
<td>PK_EXPANDED_CALL_VARIABLE_PKEY</td>
<td>expanded-call-variable-pkey</td>
</tr>
<tr>
<td>IP Endpoint Pkey</td>
<td>PK_IP_ENDPOINT_PKEY</td>
<td>ip-endpoint-pkey</td>
</tr>
<tr>
<td>Label Pkey</td>
<td>PK_LABEL_PKEY</td>
<td>label-pkey</td>
</tr>
<tr>
<td>Network VRU Script Pkey</td>
<td>PK_NETWORK_VRU_SCRIPT_PKEY</td>
<td>network-vru-script-pkey</td>
</tr>
<tr>
<td>Person Pkey</td>
<td>PK_PERSON_PKEY</td>
<td>person-pkey</td>
</tr>
<tr>
<td>Precision Attribute Pkey</td>
<td>PK_PRECISION_ATTRIBUTE_PKEY</td>
<td>precision-attribute-pkey</td>
</tr>
<tr>
<td>Precision Queue Pkey</td>
<td>PK_PRECISION_QUEUE_PKEY</td>
<td>precision-queue-pkey</td>
</tr>
<tr>
<td>Precision Queue Step Pkey</td>
<td>PK_PRECISION_QUEUE_STEP_PKEY</td>
<td>precision-queue-step</td>
</tr>
<tr>
<td>Route Pkey</td>
<td>PK_ROUTE_PKEY</td>
<td>route-pkey</td>
</tr>
<tr>
<td>Route Partition Pkey</td>
<td>PK_ROUTE_PARTITION_PKEY</td>
<td>route-partition-pkey</td>
</tr>
<tr>
<td>Service Pkey</td>
<td>PK_SERVICE_PKEY</td>
<td>service-pkey</td>
</tr>
<tr>
<td>Skill Group Pkey</td>
<td>PK_SKILLGROUP_PKEY</td>
<td>skill-group-pkey</td>
</tr>
<tr>
<td>Tenant Pkey</td>
<td>PK_TENANT_PKEY</td>
<td>tenant-pkey</td>
</tr>
<tr>
<td>User Variable Pkey</td>
<td>PK_USER_VARIABLE_PKEY</td>
<td>user-variable-pkey</td>
</tr>
</tbody>
</table>
Non Provisionable Remote Resource Pkey Map Types

These remote resource pkey map types are supported by the Resource Management Web Services for searching only.

<table>
<thead>
<tr>
<th>Resource Pkey Map Type</th>
<th>Internal Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Announcement Pkey</td>
<td>PK_ANNOUNCEMENT_PKEY</td>
</tr>
<tr>
<td>Application Gateway Pkey</td>
<td>PK_APPLICATION_GATEWAY_PKEY</td>
</tr>
<tr>
<td>Application Instance Pkey</td>
<td>PK_APPLICATION_INSTANCE_PKEY</td>
</tr>
<tr>
<td>Bucket Interval Pkey</td>
<td>PK_BUCKET_INTERVAL_PKEY</td>
</tr>
<tr>
<td>Call Manager Group Pkey</td>
<td>PK_CALL_MANAGER_GROUP_PKEY</td>
</tr>
<tr>
<td>Call Source Pkey</td>
<td>PK_CALL_SOURCE_PKEY</td>
</tr>
<tr>
<td>Campaign Pkey</td>
<td>PK_CAMPAIGN_PKEY</td>
</tr>
<tr>
<td>Category Pkey</td>
<td>PK_CATEGORY_PKEY</td>
</tr>
<tr>
<td>Chargeband Pkey</td>
<td>PK_CHARGEBAND_PKEY</td>
</tr>
<tr>
<td>Cli Pkey</td>
<td>PK_CLI_PKEY</td>
</tr>
<tr>
<td>Custom Entry Pkey</td>
<td>PK_CUSTOM_ENTRY_PKEY</td>
</tr>
<tr>
<td>Date Time Setting Pkey</td>
<td>PK_DATE_TIME_SETTING_PKEY</td>
</tr>
<tr>
<td>Device Pool Pkey</td>
<td>PK_DEVICE_POOL_PKEY</td>
</tr>
<tr>
<td>Device Target Pkey</td>
<td>PK_DEVICE_TARGET_PKEY</td>
</tr>
<tr>
<td>Dialer Pkey</td>
<td>PK DIALER_PKEY</td>
</tr>
<tr>
<td>Dial Number Plan Pkey</td>
<td>PK_DIAL_NUMBER_PLAN_PKEY</td>
</tr>
<tr>
<td>Enterprise Route Pkey</td>
<td>PK_ENTERPRISE_ROUTE_PKEY</td>
</tr>
<tr>
<td>Enterprise Service Pkey</td>
<td>PK_ENTERPRISE_SERVICE_PKEY</td>
</tr>
<tr>
<td>Gateway Function Pkey</td>
<td>PK_GATEWAY_FUNCTION_PKEY</td>
</tr>
<tr>
<td>Gateway Result Pkey</td>
<td>PK_GATEWAY_RESULT_PKEY</td>
</tr>
<tr>
<td>Gateway Server Pkey</td>
<td>PK_GATEWAY_SERVER_PKEY</td>
</tr>
<tr>
<td>ICR Instance Pkey</td>
<td>PK_ICR_INSTANCE_PKEY</td>
</tr>
<tr>
<td>Resource Pkey Map Type</td>
<td>Internal Name</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>Import Rule Pkey</td>
<td>PK_IMPORT_RULE_PKEY</td>
</tr>
<tr>
<td>IP Endpoint Button Template Pkey</td>
<td>PK_IP_ENDPOINT_BUTTON_TEMPLATE_PKEY</td>
</tr>
<tr>
<td>IP Endpoint Model Pkey</td>
<td>PK_IP_ENDPOINT_MODEL_PKEY</td>
</tr>
<tr>
<td>IVR Entry Point Pkey</td>
<td>PK_IVR_ENTRY_POINT_PKEY</td>
</tr>
<tr>
<td>IVR Module Pkey</td>
<td>PK_IVR_MODULE_PKEY</td>
</tr>
<tr>
<td>IVR Routing Target Pkey</td>
<td>PK_IVR_ROUTING_TARGET_PKEY</td>
</tr>
<tr>
<td>IVR Script Pkey</td>
<td>PK_IVR_SCRIPT_PKEY</td>
</tr>
<tr>
<td>IVR Script Node Pkey</td>
<td>PK_IVR_SCRIPT_NODE_PKEY</td>
</tr>
<tr>
<td>Logical Interface Controller Pkey</td>
<td>PK_LOGICAL_INTERFACE_CONTROLLER_PKEY</td>
</tr>
<tr>
<td>Media Class Pkey</td>
<td>PK_MEDIA_CLASS_PKEY</td>
</tr>
<tr>
<td>Media Routing Domain Pkey</td>
<td>PK_MEDIA_ROUTING_DOMAIN_PKEY</td>
</tr>
<tr>
<td>Network Trunk Group Pkey</td>
<td>PK_NETWORK_TRUNK_GROUP_PKEY</td>
</tr>
<tr>
<td>Network Vru Pkey</td>
<td>PK_NETWORK_VRU_PKEY</td>
</tr>
<tr>
<td>Object Type Pkey</td>
<td>PK_OBJECT_TYPE_PKEY</td>
</tr>
<tr>
<td>Peripheral Pkey</td>
<td>PK_PERIPHERAL_PKEY</td>
</tr>
<tr>
<td>Physical Interface Controller Pkey</td>
<td>PK_PHYSICAL_INTERFACE_CONTROLLER_PKEY</td>
</tr>
<tr>
<td>Port Pkey</td>
<td>PK_PORT_PKEY</td>
</tr>
<tr>
<td>Query Rule Pkey</td>
<td>PK_QUERY_RULE_PKEY</td>
</tr>
<tr>
<td>Rating Period Pkey</td>
<td>PK_RATING_PERIOD_PKEY</td>
</tr>
<tr>
<td>Reason Code Pkey</td>
<td>PK_REASON_CODE_PKEY</td>
</tr>
<tr>
<td>Region Pkey</td>
<td>PK_REGION_PKEY</td>
</tr>
<tr>
<td>Route Partition Pkey</td>
<td>PK_ROUTE_PARTITION_PKEY</td>
</tr>
<tr>
<td>Routing Client Pkey</td>
<td>PK_ROUTING_CLIENT_PKEY</td>
</tr>
<tr>
<td>Routing Script Pkey</td>
<td>PK_ROUTING_SCRIPT_PKEY</td>
</tr>
</tbody>
</table>
## Resource Pkey Map Type

<table>
<thead>
<tr>
<th>Resource Pkey Map Type</th>
<th>Internal Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedule Pkey</td>
<td>PK_SCHEDULE_PKEY</td>
</tr>
<tr>
<td>Scheduled Target Pkey</td>
<td>PK_SCHEDULED_TARGET_PKEY</td>
</tr>
<tr>
<td>Script Pkey</td>
<td>PK_SCRIPT_PKEY</td>
</tr>
<tr>
<td>Script Node Pkey</td>
<td>PK_SCRIPT_NODE_PKEY</td>
</tr>
<tr>
<td>Strategy Pkey</td>
<td>PK_STRATEGY_PKEY</td>
</tr>
<tr>
<td>Timeband Pkey</td>
<td>PK_TIMEBAND_PKEY</td>
</tr>
<tr>
<td>Tli Pkey</td>
<td>PK_TLI_PKEY</td>
</tr>
<tr>
<td>Trunk Pkey</td>
<td>PK_TRUNK_PKEY</td>
</tr>
<tr>
<td>Trunk Group Pkey</td>
<td>PK_TRUNK_GROUP_PKEY</td>
</tr>
<tr>
<td>Wrapup Code Pkey</td>
<td>PK_WRAPUP_CODE_PKEY</td>
</tr>
</tbody>
</table>

## Member Pkey Map Types

These member pkey map types are fully supported by the Resource Management Web Services.

<table>
<thead>
<tr>
<th>Member Pkey Map Type</th>
<th>Internal Name</th>
<th>REST Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Agent Team Member Pkey</td>
<td>PK_AGENT_AGENT_TEAM_MEMBER_PKEY</td>
<td>agent-agent-team_member-pkey</td>
</tr>
<tr>
<td>Agent Peripheral Member Pkey</td>
<td>PK_AGENT_PERIPHERAL_MEMBER_PKEY</td>
<td>agent-peripheral-member-pkey</td>
</tr>
<tr>
<td>Agent Precision Attribute Member Pkey</td>
<td>PK_AGENT_PREFERENCE_ATTRIBUTE_MEMBER_PKEY</td>
<td>agent-precision-attribute-member-pkey</td>
</tr>
<tr>
<td>Agent Skill Group Member Pkey</td>
<td>PK_AGENT_SKILLGROUP_MEMBER_PKEY</td>
<td>agent-skill-group-member-pkey</td>
</tr>
<tr>
<td>Call Type Routing Script Member Pkey</td>
<td>PK_CALL_TYPE_ROUTING_SCRIPT_MEMBER_PKEY</td>
<td>call-type-routing-script-member-pkey</td>
</tr>
<tr>
<td>Device Profile Directory Number Member Pkey</td>
<td>PK DEVICE_PROFILE_DIRECTORY_NUMBER_MEMBER_PKEY</td>
<td>device-profile-directory_number-member-pkey</td>
</tr>
</tbody>
</table>
### Member Pkey Map Type

<table>
<thead>
<tr>
<th>Internal Name</th>
<th>REST Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dialed Number Call Type Member Pkey</strong></td>
<td><code>di al ed-number-call-type-member-pkey</code></td>
</tr>
<tr>
<td><strong>IP Endpoint Directory Number Member Pkey</strong></td>
<td><code>i p-endpoi nt-di rectory-number-member-pkey</code></td>
</tr>
<tr>
<td><strong>Precision Queue Step Precision Attribute Member Pkey</strong></td>
<td><code>pre ci sion-queue-step-precision-queue-attribute-member-pkey</code></td>
</tr>
<tr>
<td><strong>Query Rule Campaign Member Pkey</strong></td>
<td><code>query-rule-campaign-member-pkey</code></td>
</tr>
<tr>
<td><strong>Route Partition Calling Search Space Member Pkey</strong></td>
<td><code>route-partition-calling-search-space-member-pkey</code></td>
</tr>
<tr>
<td><strong>Skill Group Campaign Member Pkey</strong></td>
<td><code>skill-group-campaign-member-pkey</code></td>
</tr>
<tr>
<td><strong>Skill Group Service Member Pkey</strong></td>
<td><code>skill-group-service-member-pkey</code></td>
</tr>
</tbody>
</table>

### Pkey Map Class

The `PkeyMap` class is used to map items to more than one equipment instance. If an item is associated with multiple equipment instances, there will be a pkey map associated with each equipment instance.

### Properties

The exposed properties on the `PkeyMap` class are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits from <code>Entry</code> (when <code>Type</code> is a pkey type or a member pkey type)</td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>ClusterResourceId</td>
<td><code>String</code> (N-v pair)</td>
<td>The identifier of the equipment instance which this pkey map relates to.</td>
<td>No</td>
</tr>
</tbody>
</table>
Item Pkey Class

The ItemPkey only applies to resources that are mapped to more than one equipment instance. It represents membership information from the equipment instances that this membership is mapped to. It may be created in Unified CCMP or imported into Unified CCMP.

Properties

The exposed properties on the ItemPkey class are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherit from PkeyMap (when Type is a pkey type)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internal Name</td>
<td>String (n-v pair)</td>
<td>The name of the entity on the equipment instance which this pkey map relates to.</td>
<td>Yes</td>
</tr>
<tr>
<td>Description</td>
<td>String (n-v pair)</td>
<td>The description of the entity on the equipment instance which this pkey map relates to.</td>
<td>No</td>
</tr>
</tbody>
</table>

Member Pkey Class

The MemberPkey class applies to memberships that that are mapped to more than one equipment instance. It represents membership information from the equipment instances that this membership is mapped to. It may be created in Unified CCMP or imported into Unified CCMP.

Properties

The exposed properties of the MemberPkey class are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherit From: PkeyMap (when Type is a member pkey type)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parent Id</td>
<td>Int (n-v pair)</td>
<td>This is the id of the resource which is the parent resource for the member pkey.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Using the Pkey Map Classes

You do not need to use the pkey map classes unless you need to map resources to multiple equipment instances. All the information about a single equipment mapping is available directly from the resource or member class.

Specific information about using the pkey map classes is available as follows:

- Each of the provisionable resource types described in section Provisionable Resource Types on page 55 includes information about the fields that may be located on pkey items.
- Each of the APIs described in section Web Service APIs on page 130 explains how to use pkey maps with that API.

### Field Validation

The Resource Management Web Service validates all supplied fields for correctness in terms of legal characters and length. However, validating these fields on the client before making the API call will improve the end customer experience and avoid round tripping for errors. The following table shows the common fields and their validation criteria.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Field</th>
<th>Regular Expression</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Various</td>
<td>Internal Name</td>
<td>^[a-zA-Z0-9][a-zA-Z0-9_-\s]*$</td>
<td>32</td>
</tr>
<tr>
<td>Agent</td>
<td>Extension</td>
<td>^[a-zA-Z0-9] +$</td>
<td>32</td>
</tr>
<tr>
<td>Agent</td>
<td>Peripheral Name</td>
<td>^[a-zA-Z0-9][a-zA-Z0-9_-]*$</td>
<td>32</td>
</tr>
<tr>
<td>Agent Desk Top</td>
<td>Device Port Address</td>
<td>^[a-zA-Z0-9][a-zA-Z0-9_-]*$</td>
<td>32</td>
</tr>
<tr>
<td>Dialed Number</td>
<td>Dialed Number String</td>
<td>. *</td>
<td>32</td>
</tr>
<tr>
<td>Directory Number</td>
<td>Number Pattern</td>
<td>^[X0-9]?[[[][^-^*^#]]+.$</td>
<td>32</td>
</tr>
<tr>
<td>IP Endpoint</td>
<td>Description</td>
<td>^[]a-zA-Z0-9\s$'(())^+./:+@[^_}{]-}*$</td>
<td>32</td>
</tr>
</tbody>
</table>
### System Resource Types

This section describes the types that represent Unified CCMP system resources.

**Folder Resource Item**

**Description**

The Folder resource describes a location in the folder hierarchy that is a container for other system resources or remote resources.

**Fields**

For folders only the default Resource level fields can be specified. The **FolderId** field represents the identity of the parent folder. For example the path `/Tenant1/Folder1` is identified by a name of `Folder1` and a **FolderId** of the `Tenant1` folder.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inherits From:</strong> Item</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Path</td>
<td>String</td>
<td>The path defining the location of this folder. This is a read-only field set</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>by Unified CCMP on Retrieve or Search operations only.</td>
<td></td>
</tr>
<tr>
<td>ParentId</td>
<td>Guid</td>
<td>The identifier of the folder that is the immediate parent of this folder.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>This is a read-only field set by Unified CCMP on Retrieve or Search</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>operations only.</td>
<td></td>
</tr>
<tr>
<td>PolicyId</td>
<td>Guid</td>
<td>The id of the security policy associated with the folder. This is a read-</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>only field set by Unified CCMP on Retrieve or Search operations only.</td>
<td></td>
</tr>
</tbody>
</table>
### System Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>FolderType</td>
<td>String (n-v pair)</td>
<td>The type of folder represented by this instance. One of ‘F’ for folder or ‘T’ for tenant folder. Folders of type ‘T’ are created by the Create Tenant use case only.</td>
<td>Yes</td>
</tr>
<tr>
<td>PolicyRoot</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the security policy associated with the folder is the policy root. A value of 1 indicates root policy. This is a read-only field set by Unified CCMP on Retrieve or Search operations only.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

#### Model

**Figure 3-3  Folder Relationships**

![Folder Relationships Diagram](image-url)
REST Protocol

Folder REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/folders/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Sync.</td>
<td>POST</td>
<td>201 OK</td>
<td>[Base URL]/folders</td>
</tr>
<tr>
<td>Delete</td>
<td>Sync.</td>
<td>DELETE</td>
<td>200 OK</td>
<td>[Base URL]/folders/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/folder</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/folders/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/folders</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_FOLDER&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/folders/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/folders

Group Resource Item

Description

The Group resource is used to group users to apply role based security. Groups can contain other groups.

Fields

Only the default Resource level fields can be set or read on a group resource.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From</td>
<td>Item</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### System Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Bool (n-v pair)</td>
<td>Flag indicating whether the group is enabled. Used to disable a security group.</td>
<td>No</td>
</tr>
<tr>
<td>Hidden</td>
<td>Bool (n-v pair)</td>
<td>Flag indicating whether the group is hidden. Used for system security groups that are not visible to tenants.</td>
<td>No</td>
</tr>
</tbody>
</table>

### Model

**Figure 3-4  Group Relationships**

![Group Relationships Diagram]

REST Protocol

**Group REST API Summary**

<table>
<thead>
<tr>
<th>Base URL</th>
<th>https://&lt;server&gt;:8085/resourceManagement/rest/resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO Format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Operation</td>
<td>Mode</td>
</tr>
<tr>
<td>-----------</td>
<td>------</td>
</tr>
<tr>
<td>Audit</td>
<td>Sync.</td>
</tr>
<tr>
<td>Create</td>
<td>Sync.</td>
</tr>
<tr>
<td>Delete</td>
<td>Sync.</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
</tr>
<tr>
<td>Update</td>
<td>Sync.</td>
</tr>
</tbody>
</table>

**Exceptions**
See Errors on page 18.

**Example**
POST https://Web01:8085/resourceManagement/rest/resources/groups

### User Resource Item

**Description**

The User resource is the login record for a Unified CCMP user. It defines the user’s association with roles, tasks and groups and determines what they are allowed to see and do. There are typically three classes of users:

- **Host Administrator** - can view and amend all resources on the complete platform.
- **Tenant Administrator** - can view and amend all resources that are owned by that tenant. This is the typical user account that calls the Unified CCMP Web Services.
- **Agent or Supervisor** - can view and amend all resources to which the tenant folder structure allows them, for example, Baltimore agents can only view resources in the Baltimore folder.

**Fields**

The following fields can be read or set for a user resource.
<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inherits From:</strong></td>
<td>Item</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LoginName</td>
<td>String(70)</td>
<td>The login name for an Unified CCMP user.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PassPhrase</td>
<td>String(100)</td>
<td>The passphrase for the normal user. Note: this is not required when operating</td>
<td>Depends on</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>with an external security provider.</td>
<td>ICM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>configuration</td>
</tr>
<tr>
<td>Description</td>
<td>String(500)</td>
<td>The description of this instance of the resource.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FailedAttempts</td>
<td>Int</td>
<td>This the number of failed login attempts by the user. This is a read-only</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>field only set on Retrieve or Search operations.</td>
<td></td>
</tr>
<tr>
<td>AccountLocked</td>
<td>Bool</td>
<td>A flag indicating whether the user's account has been locked or not. This is</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>a read-only field only set on Retrieve or Search operations. A value of True</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>indicates the account is locked. The default is False.</td>
<td></td>
</tr>
<tr>
<td>PassPhraseExpired</td>
<td>Bool</td>
<td>A flag indicating whether the user's passphrase has expired or not. Value of</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>true indicated pass phrase expired. Default is false.</td>
<td></td>
</tr>
<tr>
<td>PassPhraseNeverExpire</td>
<td>Bool</td>
<td>A flag indicating whether the user's passphrase will never expire. A value of</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>true indicates that the passphrase never expires. Default is false.</td>
<td></td>
</tr>
<tr>
<td>PassPhraseChangeEnabled</td>
<td>Bool</td>
<td>A flag indicating whether the user can change their passphrase or not. A</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>value of true indicates that the user can change their passphrase. Default is</td>
<td></td>
</tr>
<tr>
<td>CreateHomeFolder</td>
<td>Bool</td>
<td>Used on create only to indicate whether a home folder should be created for</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>this user.</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td>String(36)</td>
<td>The first name of the user.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### System Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>LastName</td>
<td>String(50) (n-v pair)</td>
<td>The last name of the user.</td>
<td>No</td>
</tr>
<tr>
<td>Email</td>
<td>String(50) (n-v pair)</td>
<td>The email address associated with the user. On some external security providers, this acts as an alternative primary key.</td>
<td>No</td>
</tr>
<tr>
<td>Expert</td>
<td>String (n-v pair)</td>
<td>A flag indicating when the user has expert mode enabled. A value of 1 indicates that expert mode is enabled.</td>
<td>No</td>
</tr>
<tr>
<td>TimeZone</td>
<td>String (n-v pair)</td>
<td>The time-zone in which the user is located. See section Appendix C for the list of valid time zones.</td>
<td>No</td>
</tr>
</tbody>
</table>

**Model**

*Figure 3-5*  
*User Relationships*

**REST Protocol**

**User REST API Summary**

<table>
<thead>
<tr>
<th>Base URL</th>
<th><code>https://&lt;server&gt;:8085/resourcemanagement/rest/resources</code></th>
</tr>
</thead>
<tbody>
<tr>
<td>IO Format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Operation</td>
<td>Mode Method Status URL</td>
</tr>
</tbody>
</table>
Provisionable Resource Types

This section describes the types that represent provisionable remote resources.

Agent Resource Item

Description

The Agent resource represents a customer service representative on a specific equipment instance (Peripheral) and contains the information for that agent specific to that equipment. A particular customer service representative may correspond to more than one agent resource. For example, John Smith the person has an agent record for a Voice Peripheral on which he answers phone calls and an agent record on an E-Mail Peripheral on which he answers emails.

Fields

The following fields can be read or set for Agent resource objects.
## Provisionable Resource Types

**Inherits From:** Dimension Item

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peripheral Urn</td>
<td>String(10)</td>
<td>The peripheral or equipment identifier on which the agent will be located. If -1 is supplied then Unified CCMP will pick the Peripheral automatically. Note: cannot be updated after the Create. Default: -1</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Agent Desktop Urn</td>
<td>String(10)</td>
<td>The desktop profile to be used by this agent. Default to -1 if not set. Note: can be modified using the Update call. Default: -1</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Person Urn</td>
<td>String(10)</td>
<td>The Person resource to which this agent will be associated. A Person can only have one Agent representation on any one Peripheral but can have a representation on all the relevant underlying Peripherals. For example, a Person can have an Agent Resource on a Voice Peripheral and an Agent Resource on an E-Mail Peripheral. Default: -1</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Peripheral Number</td>
<td>String(50)</td>
<td>This is the agents login ID at the switch</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Peripheral Name</td>
<td>String(50)</td>
<td>This is the name of the agent as known to the peripheral.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Supervisor</td>
<td>Bool</td>
<td>This is a flag indicating whether the agent can be a supervisor of agent teams or not. Default: false.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Element Name</td>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
<td>Pkey?</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td>AgentStateTrace</td>
<td>Bool (n-v pair)</td>
<td>This is a flag indicating whether the agent state trace data needs to be collected or not. Default: false.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DomainName</td>
<td>String(255) (n-v pair)</td>
<td>This is the DNS format of the active directory domain name. This is populated only if the agent is set as a Supervisor</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DomainLoginName</td>
<td>String(64) (n-v pair)</td>
<td>This is the domain login name. This is populated only if the agent is set as a supervisor.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DomainUserName</td>
<td>String(255) (n-v pair)</td>
<td>This is the user name. This is populated only if the agent is set as a supervisor.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DomainUserGuid</td>
<td>String(36) (n-v pair)</td>
<td>This is the unique identifier of the user in the active directory. This is populated only if the agent is set as a supervisor.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DomainPassPhrase</td>
<td>String(50) (n-v pair)</td>
<td>For future use. This is the pass phrase for the user on the Active Directory</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>UserDeletable</td>
<td>Bool (n-v pair)</td>
<td>Indicates if the item can be deleted by end users as opposed to the Unified CCMP background processes. Note: used for Unified CCE parent/child configurations only.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>UserGroupId</td>
<td>String (n-v pair)</td>
<td>The User Group ID for this Agent on Unified CCE.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int (n-v pair)</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
# Model

## Figure 3-6  Agent Relationships

![Agent Relationships Diagram](image)

## REST Protocol

### Agent REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/agents/&lt;d&gt;,&lt;d&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;…</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/agents</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/agents/&lt;d&gt;,&lt;d&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/agent</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationId&gt;</td>
</tr>
</tbody>
</table>
Agent REST API Summary

<table>
<thead>
<tr>
<th>Retrieve</th>
<th>Sync.</th>
<th>GET</th>
<th>200 OK</th>
<th>[Base URL]/agents/&lt;d&gt;,&lt;d&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/agents</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_AGENT &amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/agents/&lt;d&gt;,&lt;d&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/agents

Agent Desktop Resource Item

Description
The Agent Desktop resource represents the options available to an agent, typically via their CTI desktop.

This resource has an optional resource membership to a dialed number which represents the Ring No Answer Dialed Number.

Fields
The following fields can be read or set for Agent Desktop resources.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey</th>
</tr>
</thead>
<tbody>
<tr>
<td>AvailableAfter Incoming</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether to consider an agent to be available after handling an incoming call.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>AvailableAfter Outgoing</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether to consider an agent to be available after handling an outbound call.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Element Name</td>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
<td>Pkey</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>------</td>
</tr>
<tr>
<td>WrapupDataIncoming</td>
<td>Int (n-v pair)</td>
<td>A flag indicating whether the agent is allowed or required to enter wrap-up data after handling an incoming call. This takes the values 0: Required; 1: Optional; 2: Not allowed.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>WrapupDataOutgoing</td>
<td>Int (n-v pair)</td>
<td>A flag indicating whether the agent is allowed or required to enter wrap-up data after handling an outbound call. This takes the values 0: Required; 1: Optional; 2: Not allowed.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>AutoAnswerEnabled</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the calls to the agent are automatically answered.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>IdleReasonRequired</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent should enter a reason before entering the Idle state.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>LogoutNonActivityTime</td>
<td>Int (n-v pair)</td>
<td>The number of seconds after which the agent will be logged out automatically if there is no activity at the agent desktop. This value is between 10 and 7200.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>LogoutReasonRequired</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent is required to enter a reason before logging out.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>SupervisorCallsAllowed</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can initiate supervisor assisted calls.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>SupervisorAssistCallMethod</td>
<td>Int (n-v pair)</td>
<td>A value that indicates whether system would create a consultative or a blind conference call for the supervisor assistance request. This takes the values 0: Consultative; 1: Blind conference.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Element Name</td>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
<td>Pkey</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>------</td>
</tr>
<tr>
<td>EmergencyCallMethod</td>
<td>Int (n-v pair)</td>
<td>A value that indicates whether system would create a consultative or a blind conference call for the emergency call request. This takes the values 0: Consultative; 1: Blind conference.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>AutoRecordOnEmergency</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether to automatically record or not to record when an emergency call request is started.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>AgentToAgentCalls</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether calls to other agents are allowed.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>OutboundAccessInternational</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can initiate international calls.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>OutboundAccessPublicNetwork</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can initiate calls through public network.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>OutboundAccessPrivateNetwork</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can initiate calls through private network.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>OutboundAccessOperatorAssisted</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can initiate operator assisted calls.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>OutboundAccessPBX</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can initiate outbound PBX calls.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>NonACDCallsAllowed</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can place or handle non-ACD calls.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>QualityRecordingRate</td>
<td>Int (n-v pair)</td>
<td>A value in seconds indicating how frequently calls to the agent are recorded.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Element Name</td>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
<td>Pkey</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>------</td>
</tr>
<tr>
<td>RecordingMode</td>
<td>Int (n-v pair)</td>
<td>A value that indicates whether the call requests are routed through Unified CM. Value 0 indicates that the calls do not get routed through Unified CM.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>WorkModeTimer</td>
<td>Int (n-v pair)</td>
<td>A value indicating the wrap-up time out in seconds. This value is between 1 and 7200.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>RingNoAnswerTime</td>
<td>Int (n-v pair)</td>
<td>A value indicating the number of seconds a call may ring at the agent’s station before it is redirected. This value is between 1 and 120.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>SilentMonitor WarningMessage</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether a warning message will appear on the agent’s desktop when silent monitoring is started.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>SilentMonitor AudibleIndication</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether an audio click will sound when silent monitoring is started.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DefaultDevice PortAddress</td>
<td>String (n-v pair)</td>
<td>A value to override the default port address of the agent’s telephony device.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>AgentCanSelectGroup</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether the agent can select which group they are logged in to.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>RemoteAgentType</td>
<td>Int (n-v pair)</td>
<td>A value indicating how mobile agents who use this dial plan are handled. This takes the values 0: Local agent, no remote access; 1: Use call by call mobile agent routing; 2: Use nailed connection for mobile agents; 3: Agent chooses at login.</td>
<td>No</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoteLoginWithoutDesktop</td>
<td>Bool (n-v pair)</td>
<td>A flag indicating whether mobile agents are allowed to login without a desktop.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int (n-v pair)</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

### Model

*Figure 3-7 Agent Desktop Relationships*

![Agent Desktop Relationships Diagram]

### REST Protocol

**Agent Desktop REST API Summary**

<table>
<thead>
<tr>
<th>Base URL</th>
<th>IO Format</th>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>https://&lt;server&gt;:8085/resourcemanagement/rest/resources</td>
<td>XML or JSON</td>
<td>Mode</td>
<td>Method</td>
<td>Status</td>
<td>URL</td>
<td></td>
</tr>
</tbody>
</table>

Web Services Reference for Cisco Unified Contact Center Management Portal

August 2015
Agent Desktop REST API Summary

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/agent-desktops/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/agent-desktops</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/agent-desktops/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/agent-desktop</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/agent-desktops/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/agent-desktops</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_AGENT DESKTOP&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/agent-desktops/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/agent-desktops

Agent Team Resource Item

Description
The Agent Team resource represents a specific collection of agent resources, usually with an assigned supervisor. Call routing does not use this resource type. It is used for line management and performance management. The Agent team is linked to a specific equipment peripheral to ensure that agents on an equipment instance are grouped together. For other grouping types use a folder resource. For the Cisco UCCE call routing platform an agent is typically only a member of one agent team.

Fields
The following fields can be read or set for Agent Team resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From</td>
<td>DimensionItem</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>PeripheralUrn</td>
<td>String(10) (n-v pair)</td>
<td>The peripheral or equipment identifier on which the agent team will be located. If -1 is supplied then Unified CCMP will pick the Peripheral automatically. Note: cannot be updated after the Create.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>DialedNumberUrn</td>
<td>String(10) (n-v pair)</td>
<td>The default dialed number associated with this agent team.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int        (n-v pair)</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

**Model**

*Figure 3-8 Agent Team Relationships*
REST Protocol

Agent Team REST API Summary

<table>
<thead>
<tr>
<th>Base URL</th>
<th>https://&lt;server&gt;:8085/resourcemanagement/rest/resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO Format</td>
<td>XML or JSON</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/agent-teams/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/agent-teams</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/agent-teams,/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/agent-team</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/agent-teams,/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/agent-teams</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_AGENT_TEAM&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/agent-teams,/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/agent-teams

Call Type Resource Item

Description
The Call Type resource represents a classifier applied to a specific class of calls. This allows the treatment of calls to be handled by a specific routing script; typically after the call has been segmented by IVR treatment or simple caller enter digits.

Fields
The following fields can be read or set for Call Type resources:
### Element Name | Data Type | Description | Required? | Pkey?
---|---|---|---|---
**Inherits From:** DimensionItem

ServiceLevelThreshold | Int (n-v pair) | The service level threshold for the Call Type. Possible values are: 0: None (default) 1: Use global settings 2 or greater: Threshold value in seconds | No | ✓

ServiceLevelType | Short (n-v pair) | This is the type that defines how service levels are calculated for the Call Type. Defaults to 0. This takes the values: 0: Default 1: Ignore abandoned calls 2: Abandoned calls have negative impact 3: Abandoned calls have positive impact. | No | ✓

DepartmentBizUrn | Int (n-v pair) | Foreign key to the department that this resource is associated with, or -1 if there is no associated department. | No |
**Figure 3-9  Call Type Relationships**

**REST Protocol**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/call-types/&lt;d&gt;,&lt;d&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;todate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/call-types</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/call-types/&lt;d&gt;,&lt;d&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/call-type</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/call-types/&lt;d&gt;,&lt;d&gt;</td>
</tr>
</tbody>
</table>

**Call Type REST API Summary**

- **Base URL**: https://<server>:8085/resourcemanagement/rest/resources
- **IO Format**: XML or JSON
### Call Type REST API Summary

<table>
<thead>
<tr>
<th>Action</th>
<th>Mode</th>
<th>Method</th>
<th>Status Code</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/call-types</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_CALL_TYPE&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/call-types/ &lt;d&gt;, &lt;d&gt;</td>
</tr>
</tbody>
</table>

**Exceptions**: See Errors on page 18.

**Example**: 

```
POST https://Web01:8085/resourcemanagement/rest/resources/call-types
```

### Calling Search Space Resource Item

**Description**

The CallingSearchSpace resource represents a specific collection of Route Partitions that are considered before an IP call can be made.

**Fields**

The following fields can be read or set for Calling Search Space resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From:</td>
<td></td>
<td>DimensionItem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clause</td>
<td>String(500) (n-v pair)</td>
<td>CSV list of Route Partition Ids</td>
<td>Yes</td>
<td>✓</td>
</tr>
</tbody>
</table>
Model

Figure 3-10  Calling Search Space Relationships

REST Protocol

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/calling-search-spaces/&lt;d&gt;, audits? fromDate=fromDate&gt;&amp; toDate=toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/calling-search-spaces</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/calling-search-spaces/&lt;d&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/calling-search-space</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
</tbody>
</table>
Chapter 3: Resource Management Web Service

Provisionable Resource Types

Calling Search Space REST API Summary

<table>
<thead>
<tr>
<th>Retrieve</th>
<th>Sync.</th>
<th>GET</th>
<th>200 OK</th>
<th>[Base URL]/calling-search-spaces/ &lt;d&gt;, &lt;d&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/calling-search-spaces</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_CALLLING_SEARCH_SPACE&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/calling-search-spaces/ &lt;d&gt;, &lt;d&gt;</td>
</tr>
</tbody>
</table>

Exceptions

See Errors on page 18.

Example

POST https://Web01:8085/resourcemanagement/rest/resources/calling-search-spaces

Department Resource Item

Description

The Department resource is a way of grouping resources according to a business need.

Fields

The following fields can be read or set for Department resources:

Table 3-4  Department Resource Item Fields

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

No further fields.
Chapter 3: Resource Management Web Service

Provisionable Resource Types

Model

Figure 3-11 Department Relationships

REST Protocol

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/department/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromDate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/departments</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/departments/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/department</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/departments/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/departments</td>
</tr>
</tbody>
</table>

Device Profile REST API Summary

| Base URL | https://<server>:8085/resourcemanagement/rest/resources |
| IO Format | XML or JSON |
Device Profile REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Method</th>
<th>Status Code</th>
<th>URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]? queryString=type%3aIT_DEPARTMENT&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/departments/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions

See Errors on page 18.

Example

POST https://Web01:8085/resourcemanagement/rest/resources/departments

Device Profile Resource Item

Description

The DeviceProfile represents a collection of fields that are associated with a particular device.

Fields

The following fields can be read or set for Device Profile resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherit From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPEndpointButtonTemplateUrn</td>
<td>String</td>
<td>Parent resource that must be supplied during the create process. Note: this is only applicable to devices which support button templates.</td>
<td>Yes, if the device supports button templates, otherwise no.</td>
<td></td>
</tr>
<tr>
<td>ProductCodeBizUrn</td>
<td>Int</td>
<td>Foreign key to a product code. Code type: CG_DEVI_CE_PRODUCT</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>ProtocolCodeBizUrn</td>
<td>Int</td>
<td>Foreign key to a protocol code. Code type: CG_DEVI_CE_PROTOCOL</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>TemplateData</td>
<td>Xml</td>
<td>The template attributes that will be used as a default for any IP Endpoint Creation</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
Model

**Figure 3-12 Device Profile Relationships**

REST Protocol

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/device-profiles/&lt;d&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;…</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/device-profiles</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/device-profiles/&lt;d&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/device-profile</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/destinationId</td>
</tr>
</tbody>
</table>
### Device Profile REST API Summary

<table>
<thead>
<tr>
<th>Action</th>
<th>Sync/Async</th>
<th>Method</th>
<th>Status Code</th>
<th>URL Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieve</td>
<td>Sync</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/device-profiles/ &lt;d&gt;, &lt;d&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/device-profiles</td>
</tr>
<tr>
<td>Search</td>
<td>Sync</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_DEVICE_PROFILE &amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/device-profiles/ &lt;d&gt;, &lt;d&gt;</td>
</tr>
</tbody>
</table>

**Exceptions**
See Errors on page 18.

**Example**
POST https://Web01:8085/resourcemanagement/rest/resources/device-profiles

### Dialed Number Resource Item

**Description**
The DialedNumber resource represents the entry point into the contact center product set. The call can be pre-screened and segmented using IVR or go straight through to an agent queuing solution or a combination of both. Care is needed with this resource type since many organizations have only a single 01800 dialed number entry point and invalid removal of this can make many agents idle.

**Fields**
The following fields can be read or set for Dialed Number resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RoutingClientUrn</td>
<td>String(10)      (n-v pair)</td>
<td>The routing client that services this dialed number.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>MediaRoutingDomainUrn</td>
<td>String(10)      (n-v pair)</td>
<td>The media routing domain that services this dialed number.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>DialedNumber</td>
<td>String(32)      (n-v pair)</td>
<td>This is a string indicating the dialable number for the directory number. This can be the actual numeric number or a route pattern with wildcards.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

#### Permit Application Routing
- **Data Type**: Bool (n-v pair)
- **Description**: Flag determining whether application routing is allowed on this dialed number.
- **Required?**: No
- **Pkey?**: Yes

#### Self Care Enabled
- **Data Type**: Bool (n-v pair)
- **Description**: Flag indicating whether self-care is enabled for this dialed number.
- **Required?**: No

#### Department Biz Urn
- **Data Type**: Int (n-v pair)
- **Description**: Foreign key to the department that this resource is associated with, or -1 if there is no associated department.
- **Required?**: No

---

**Model**

#### Figure 3-13 Dialed Number Relationships

![Dialed Number Relationships Diagram](image)

---

**REST Protocol**

#### Dialed Number REST API Summary

<table>
<thead>
<tr>
<th>Base URL</th>
<th>https://&lt;server&gt;:8085/resourcemanagement/rest/resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO Format</td>
<td>XML or JSON</td>
</tr>
</tbody>
</table>
## Dialed Number REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>![Base URL]/dialed-numbers/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>![Base URL]/dialed-numbers</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>![Base URL]/dialed-numbers/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>![Base URL]/meta/dialed-number</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>![Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>![Base URL]/dialed-numbers/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>![Base URL]/members/dialed-numbers</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>![Base URL]?queryString=type%3aIT_DIALED_NUMBER&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>![Base URL]/dialed-numbers/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

### Exceptions
See Errors on page 18.

### Example
POST https://Web01:8085/resourcemanagement/rest/resources/dialed-numbers

### Directory Number Resource Item

#### Description
The DirectoryNumber resource represents an extension number or number pattern on a switch. This type is only created by Unified CCMP when Unified CCMP is the only provisioning tool. When other provisioning tools are used then Unified CCMP detects new instances of this type by reading the switch directly as part of the import process.

#### Fields
The following fields can be read or set for Directory Number resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Element Name</td>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
<td>Pkey?</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td>DirectoryNumberOrPattern</td>
<td>String(50)</td>
<td>This is a string indicating the dial-able number for the directory number. This can be the actual numeric number or a route pattern with wildcards.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>IPCCEnabled</td>
<td>Bool</td>
<td>This is a flag indicating whether the directory number is IPCC enabled. For example; in a Cisco UCCE deployment model this value is set if there is a matching device target on the Unified CCE associated with the Unified CM to which this directory number belongs. This matching is done based on the value in the CONFIG_PARAM column in TB_DIM_DIRECTORY_NUMBER and the DIRECTORY_NUMBER_OR_PATTERN in for this record. The default value is false (the directory number is not IPCC-enabled).</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>RoutePartitionBizUrn</td>
<td>Int</td>
<td>This is the route partition associated with the directory number.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
### Model

**Figure 3-14 Directory Number Relationships**

REST Protocol

<table>
<thead>
<tr>
<th>Directory Number REST API Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base URL</strong></td>
</tr>
<tr>
<td><strong>IO Format</strong></td>
</tr>
<tr>
<td><strong>Operation</strong></td>
</tr>
<tr>
<td>Audit</td>
</tr>
<tr>
<td>Create</td>
</tr>
<tr>
<td>Delete</td>
</tr>
<tr>
<td>Describe</td>
</tr>
<tr>
<td>Move</td>
</tr>
<tr>
<td>Retrieve</td>
</tr>
</tbody>
</table>
Chapter 3: Resource Management Web Service

Provisionable Resource Types

Directory Number REST API Summary

<table>
<thead>
<tr>
<th>Save</th>
<th>Sync.</th>
<th>POST</th>
<th>200 OK</th>
<th>[Base URL]/members/directory-numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_DIRECTORY_NUMBER &amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/directory-numbers/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/directory-numbers

Enterprise Skillgroup Resource Item

Description
The EnterpriseSkillgroup represents a specific collection of skillgroups.

Fields
The following fields can be read or set for Enterprise Skillgroup resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From:</td>
<td></td>
<td>Di mension Item</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int (n-v pair)</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
Model

**Figure 3-15  Enterprise Skillgroup Relationships**

REST Protocol

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/enterprise-skillgroups/&lt;id&gt;:&lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/enterprise-skillgroups</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/enterprise-skillgroups/&lt;id&gt;:&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/enterprise-skillgroup</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
</tbody>
</table>
Chapter 3: Resource Management Web Service

Provisionable Resource Types

Enterprise Skillgroup REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Sync/Async</th>
<th>HTTP Method</th>
<th>Status Code</th>
<th>URI Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieve</td>
<td>Sync</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/enterprise-skillgroups/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/enterprise-skillgroups</td>
</tr>
<tr>
<td>Search</td>
<td>Sync</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/?queryString=type%3aIT_ENTERPRISE_DIRECT &amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/enterprise-skillgroups/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions

See Errors on page 18.

Example

POST https://Web01:8085/resourcemanagement/rest/resources/enterprise-skillgroups

Expanded Call Variable Resource Item

Description

The Expanded Call Variable resource is used to hold scripting and other data variables.

Fields

The following fields can be read or set for Expanded Call Variable resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECCArray</td>
<td>Bool (n-v pair)</td>
<td>This is a flag indicating whether the call variable is an array. Default: false.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>VariableEnabled</td>
<td>Bool (n-v pair)</td>
<td>This is a flag indicating whether the call variable is enabled. Default: false.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>GeoTelProvided</td>
<td>Bool (n-v pair)</td>
<td>This is a flag indicating whether the call variable is provided by Cisco. Default: false.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Element Name</td>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
<td>Pkey?</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td>Maxi numArraySi ze</td>
<td>Short (n-v pair)</td>
<td>This value indicates the number of elements in the array if the call variable is set as an array. The values are between 1 and 255. Default: 0</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Maxi numLength</td>
<td>Short (n-v pair)</td>
<td>This value indicates the maximum length for the call variable. This value is between 1 and 210. Default: 1</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Persistent</td>
<td>Bool (n-v pair)</td>
<td>This is a flag indicating whether the call variable is persistent or not. Default: true.</td>
<td>No</td>
<td>✓</td>
</tr>
</tbody>
</table>

**Model**

*Figure 3-16  Expanded Call Variable Relationships*
REST Protocol

### Expanded Call Variable REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/expanded-call-variables/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromDate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/expanded-call-variables</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/expanded-call-variables/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/label</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationId&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/expanded-call-variables/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/expanded-call-variables</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type@IT_EXPANDED_CALL_VARIABLE &amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/expanded-call-variables/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

#### Exceptions
See Errors on page 18.

#### Example

POST https://Web01:8085/resourcemanagement/rest/resources/expanded-call-variables

### IP Endpoint Resource Item

#### Description

The IPEndpoint resource represents the termination point of any IP stream. This includes devices such as phones as well as other IP sinks. This API is only used when Unified CCMP is the only provisioning tool; when there are other provisioning tools, Unified CCMP detects this type by reading the switch directly.

#### Fields

The following fields can be read or set on IP Endpoint Resource objects.
<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peripheral Urn</td>
<td>String(10) (n-v pair)</td>
<td>The peripheral or equipment identifier on which the IP Endpoint will be located. Note: cannot be updated after the Create.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>DevicePoolUrn</td>
<td>String(10) (n-v pair)</td>
<td>The device pool in which this IP Endpoint will be located. Note: cannot be modified using the Update call.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>CallingSearchSpaceUrn</td>
<td>String(10) (n-v pair)</td>
<td>The Calling Search Space in which this IP Device will be located.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>IPEndPointButtonTemplateUrn</td>
<td>String(10) (n-v pair)</td>
<td>The button template to be associated with this device. Note: this is only applicable to devices which support button templates.</td>
<td>Yes, if the device supports button templates, no otherwise.</td>
<td></td>
</tr>
<tr>
<td>ProductCodeBizUrn</td>
<td>String(10) (n-v pair)</td>
<td>Foreign key to a product code. Code type: CG.DEVICE.PRODUCT. Cannot be updated after creation.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>ProtocolCodeBizUrn</td>
<td>String(10) (n-v pair)</td>
<td>Foreign key to a protocol code. Code type: CG.DEVICE.PROTOCOL. Cannot be updated after creation.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>MACAddress</td>
<td>String(17) (n-v pair)</td>
<td>The MAC address of the device.</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>TemplateData</td>
<td>Xml (n-v pair)</td>
<td>The template data to be used with creating this IP device. Cannot be updated after creation.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>ExtensionMobilityEnabled</td>
<td>Bool (n-v pair)</td>
<td>Flag indicating whether the extension mobility is to be used with this IP Device.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

**Model**

*Figure 3-17  IP Endpoint Relationships*

**REST Protocol**

<table>
<thead>
<tr>
<th><strong>IP Endpoint REST API Summary</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base URL</strong></td>
</tr>
<tr>
<td><strong>IO Format</strong></td>
</tr>
</tbody>
</table>
## IP Endpoint REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Metho d</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/ip-endpoints/&lt;id&gt;, &lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;todate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/ip-endpoints</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/ip-endpoints/&lt;id&gt;, &lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/ip-endpoint</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/ip-endpoints/&lt;id&gt;, &lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/ip-endpoints</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_ENDPOINT&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/ip-endpoints/&lt;id&gt;, &lt;id&gt;</td>
</tr>
</tbody>
</table>

### Exceptions

See Errors on page 18.

### Example

**POST** https://Web01:8085/resourcemanagement/rest/resources/ip-endpoints

### Auto Selection of Device Pool

When creating a phone through the web service and -1 is specified for the `DevicePoolUrn`, Unified CCMP uses the following logic to automatically select a device pool to associate with the new phone:

1. Lookup the call manager time group to find a time group that corresponds to the user’s time zone. This lookup is performed first using the standard name for the time zone, otherwise the time zone offset and daylight savings information is used.

2. Lookup the call manager group as follows:
   a. If a phone already exists that is associated (via the device pool) to a call manager group with spare capacity (observing the 250 phone capacity limit), then select the call manager group of the existing phone.
   b. Otherwise, select the call manager group with the lowest available capacity but with enough space for a further 20 (configurable) phones.
3. Select the Device Pool associated with the chosen call manager group and time group.

The following errors may be reported:

- Unable to locate a call manager group with spare capacity.
- Unable to find a time group for the user’s time zone.

The capacity limit of 250 phones per call manager group is configured by the data stored in TB_ADM_CAPACITY.

**Label Resource Item**

**Description**

The Label resource represents a string that is passed to a routing client for each network target.

**Fields**

The following fields can be read or set for Label resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From:</td>
<td>DimensionItem</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RoutingClientUrn</td>
<td>String (n-v pair)</td>
<td>Parent resource that must be supplied during the create process.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Label Type</td>
<td>short (n-v pair)</td>
<td>This value indicates the type of the label. This takes the values 0: Normal; 1: DNIS override; 2: Busy; 3: Ring; 4: Post-Query; 5: Resource.</td>
<td>Yes</td>
<td>✓</td>
</tr>
</tbody>
</table>
Chapter 3: Resource Management Web Service

Provisionable Resource Types

Model

Figure 3-18 Label Relationships

REST Protocol

Label REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/labels/&lt;d&gt;/audits?fromDate=&lt;fromDate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/labels</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/labels/&lt;d&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/label</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationId&gt;</td>
</tr>
</tbody>
</table>

Base URL: https://<server>:8085/resourcemanagement/rest/resources

IO Format: XML or JSON
Label REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Action</th>
<th>Method</th>
<th>Status Code</th>
<th>URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/labels/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/labels</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_LABEL&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/labels/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions  See Errors on page 18.

Example  POST https://Web01:8085/resourcemangement/rest/resources/labels

Network VRU Script Resource Item

Description  The NetworkVruScript resource represents a script run on a Network VRU to handle a call.

Fields  The following fields can be read or set for Network VRU Script resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>VruScriptName</td>
<td>String (n-v pair)</td>
<td>This is the name of the script on the VRU.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Interruptible</td>
<td>Bool (n-v pair)</td>
<td>This is a flag indicating whether the script can be interrupted. Default: False.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Overridable</td>
<td>Bool (n-v pair)</td>
<td>This is flag indicating whether the VRU script can override its interruptible flag. Default: False.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>ConfigParam</td>
<td>String (n-v pair)</td>
<td>This is string that is sent to the VRU for initialization.</td>
<td>No</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timeout</td>
<td>Int (n-v pair)</td>
<td>This is the number of seconds for which the software will wait for a response from the routing client after directing it to run the script. Default: 0.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int (n-v pair)</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

**Model**

*Figure 3-19  Network VRU Script Relationships*

![Network VRU Script Relationships Diagram](image)

**REST Protocol**

*Network VRU Script REST API Summary*

<table>
<thead>
<tr>
<th>Base URL</th>
<th><code>https://&lt;server&gt;:8085/resourcemanagement/rest/resources</code></th>
</tr>
</thead>
</table>
### Network VRU Script REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td><code>[Base URL]/network-vru-scripts/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;todate&gt;...</code></td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td><code>[Base URL]/network-vru-scripts</code></td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td><code>[Base URL]/network-vru-scripts/&lt;id&gt;,&lt;id&gt;</code></td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td><code>[Base URL]/meta/network-vru-script</code></td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td><code>[Base URL]/&lt;destinationid&gt;</code></td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td><code>[Base URL]/network-vru-scripts/&lt;id&gt;,&lt;id&gt;</code></td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td><code>[Base URL]/members/network-vru-scripts</code></td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td><code>[Base URL]?queryString=type%3aN_NETWORK_VRU_SCRIPT&amp;max%3a10</code></td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td><code>[Base URL]/network-vru-scripts/&lt;id&gt;,&lt;id&gt;</code></td>
</tr>
</tbody>
</table>

**Exceptions**
See Errors on page 18.

**Example**
POST https://Web01:8085/resourcemanagement/rest/resources/network-vru-scripts

### Person Resource Item

**Description**

The Person resource represents any person resource on the system, not only customer service representatives but also managers, end customers (people not registered on switch) and is often used for high speed data dip for CRM style data directed routing and IVR services.

**Fields**

The following fields can be read or set for Person resources.
<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherit From:</td>
<td>DimensionItem</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Title</td>
<td>String(20)</td>
<td>This is the title for the person. This is not a constrained vocabulary.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Gender</td>
<td>String(20)</td>
<td>This is the title for the person. This is not a constrained vocabulary.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DOB</td>
<td>DateTime</td>
<td>This is the date of birth of the person.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DOD</td>
<td>DateTime</td>
<td>This is the date of death of the person.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td>String(50)</td>
<td>This is the first name of the person.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>LastName</td>
<td>String(50)</td>
<td>This is the last name of the person.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Address</td>
<td>String(255)</td>
<td>This is the communication address of the person if specified.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>EmployeeType</td>
<td>String(20)</td>
<td>The tenant specific classification if this person is an employee.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>GovernmentID</td>
<td>String(50)</td>
<td>The country specific identifier for this person for example, Social Security number.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>EmployeeID</td>
<td>String(50)</td>
<td>The company specific identifier for this person for example, payroll number.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Grade</td>
<td>Short</td>
<td>The grade or rank of this person within the company.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Competence</td>
<td>Short</td>
<td>The competence level of this person within the specified grade.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>UserID</td>
<td>Guid</td>
<td>The primary identifier of this user</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>LoginName</td>
<td>String(50)</td>
<td>This is the login name associated with the person. This is unique</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>PassPhrase</td>
<td>String(50) (n-v pair)</td>
<td>This is the pass phrase used by the person with the associated login name.</td>
<td>Depends on ICM configuration</td>
<td></td>
</tr>
<tr>
<td>LoginEnabled</td>
<td>Bool (n-v pair)</td>
<td>This field indicates whether login for the person is enabled or not.</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int (n-v pair)</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

#### Model

**Figure 3-20  Person Relationships**

![Person Relationships Diagram]

#### REST Protocol

**Person REST API Summary**

| Base URL | https://<server>:8085/resourcemanagement/rest/resources |
Person REST API Summary

<table>
<thead>
<tr>
<th>IO Format</th>
<th>XML or JSON</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation</td>
<td>Mode</td>
</tr>
<tr>
<td>Audit</td>
<td>Sync.</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
</tr>
<tr>
<td>Exceptions</td>
<td>See Errors on page 18.</td>
</tr>
<tr>
<td>Example</td>
<td>POST <a href="https://Web01:8085/resourcemanagement/rest/resources/persons">https://Web01:8085/resourcemanagement/rest/resources/persons</a></td>
</tr>
</tbody>
</table>

Precision Attribute Resource Item

Description

The PrecisionAttribute resource represents the attributes that may belong to an Agent. It is used for precision queue based call routing.

Fields

The following fields can be read or set for Precision Attribute resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From</td>
<td>Dimension Item</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttributeDataType</td>
<td>Int (n-v pair)</td>
<td>Use this parameter to assign a data type to the attribute using the following values: 3 = Boolean, 4 = Proficiency (special form of Integer)</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>MinimumValue</td>
<td>String (n-v pair)</td>
<td>For Proficiency data types, valid values are 1-10. Otherwise, NULL</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>MaximumValue</td>
<td>String (n-v pair)</td>
<td>For Proficiency data types, valid values are 1-10. Otherwise, NULL</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DefaultValue</td>
<td>String (n-v pair)</td>
<td>Use this parameter to specify a default value for the attribute when assigned to an agent if no explicit value is provided. For Boolean data types, valid default values are True and False. For Proficiency data types, valid default values are 1-10</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>AppearsOnDesktop</td>
<td>Bool (n-v pair)</td>
<td>For future use.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>SettableByAgent</td>
<td>Bool (n-v pair)</td>
<td>For future use.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int (n-v pair)</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
Model

![Diagram showing Agent, Folder, and Precision Attribute relationships]

**Figure 3-21** Precision Attribute Relationships

### REST Protocol

**Precision Attribute Resource REST API Summary**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td><code>[Base URL]/precision-attributes</code></td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td><code>[Base URL]/precision-attributes/&lt;d&gt;,&lt;d&gt;</code></td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td><code>[Base URL]/meta/precision-attribute</code></td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td><code>[Base URL]/&lt;destinationid&gt;</code></td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td><code>[Base URL]/precision-attributes/&lt;d&gt;,&lt;d&gt;</code></td>
</tr>
</tbody>
</table>
### Precision Attribute Resource REST API Summary

<table>
<thead>
<tr>
<th>Save</th>
<th>Sync.</th>
<th>POST</th>
<th>200 OK</th>
<th>[Base URL]/members/precision-attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_PRECISION_ATTRIBUTE&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/precision-attributes/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

#### Exceptions
See Errors on page 18.

#### Example
POST https://Web01:8085/resourcemanagement/rest/resources/precision-attributes

### Precision Queue Step Resource Item

#### Description

The PrecisionQueueStep resource is a set of ordered steps that apply to a Precision Queue.

A Precision Queue Step cannot exist on its own: it is always associated with a single Precision Queue. A Precision Queue Step must also be associated with at least one Precision Attribute.

#### Creating Precision Queue Steps

When a Precision Queue is first created, at least one Precision Queue Step and the associated memberships must be specified and created at the same time.

When you create a Precision Queue Step independently of a Precision Queue, you must also:

- specify an existing parent Precision Queue for the Precision Queue Step (the Precision Queue Step Precision Queue Member item that links the queue and the step will be created automatically when the step is created)

- create one or more Precision Queue Step Precision Attribute Member items that link the step that you are creating with the required attributes.

Depending on the position in the queue where you have added the step, you may also need to adjust the properties of the existing steps to maintain the correct numbering sequence and contents.

#### Deleting Precision Queue Steps

When you delete a Precision Queue Step independently of a Precision Queue, the following related items are deleted automatically:

- the member item that links the step with the queue

- any member items that link attributes with the step that you are about to delete.
Depending on the position in the queue from where you have deleted the step, you may also need to adjust the properties of the remaining steps to maintain the correct numbering sequence.

You cannot delete the last Precision Queue Step in a Precision Queue unless you are also deleting the Precision Queue itself.

**Updating Precision Queue Steps**

You can use the `Update()` method to update a Precision Queue Step, but note that you cannot change the value specified in the `PrecisionQueueUrn` field. If you want to move a step to a different queue, you must delete the original step and recreate it in the new queue.

**Adding Precision Attributes to Precision Queue Steps**

To add a Precision Attribute to a Precision Queue Step, create a Precision Queue Step Precision Attribute Member item that links the step to the required attribute. A step can have up to 10 associated attributes.

**Deleting Attributes from Precision Queue Steps**

To delete a Precision Attribute from a Precision Queue Step, delete the Precision Queue Step Precision Attribute Member item that links the step to the attribute.

You cannot delete the last Precision Queue Step Precision Attribute Member in a Precision Queue Step unless you delete the step itself.

**Fields**

The following fields can be read or set for Precision Queue Step resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>StepOrder</td>
<td>Int</td>
<td>The order of rows for a precision queue step. This value must start at 1 (zero is invalid) and increment by 1 for each subsequent step.</td>
<td>Yes</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>PrecisionQueueUrn</code></td>
<td><code>String (n-v pair)</code></td>
<td>The id of the parent Precision Queue.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td><code>WaitTime</code></td>
<td><code>Int (n-v pair)</code></td>
<td>A Wait time to apply before proceeding to the next step (in seconds). The value must be 0 or greater for all steps, excluding the last step. The value for the last step defaults to -1. With a value of -1, the system waits until an agent is available to take the call</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td><code>ConsiderIf</code></td>
<td><code>String (n-v pair)</code></td>
<td>Consider If expression which must be met in order to execute a particular step. The length of this field is 255 characters. Objects used in the expression are case-sensitive. This field must be null for the last step in a Precision Queue.</td>
<td>Yes</td>
<td>✓</td>
</tr>
</tbody>
</table>

**Model**

![Diagram of Precision Queue, Folder, Precision Queue Step, Precision Queue Attribute, and Statuses (Provisioned, Commissioned, Existing)]
**REST Protocol**

### Precision Queue Step Resource REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/precision-queue-steps/\langle d\rangle, \langle d\rangle/audits?FromDate=\langle fromdate\rangle&amp;toDate=\langle todate\rangle...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/precision-queue-steps</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/precision-queue-steps/\langle d\rangle, \langle d\rangle</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/precision-queue-step</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/destination/\langle d\rangle</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/precision-queue-steps/\langle d\rangle, \langle d\rangle</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/precision-queue-steps</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_PRECISION_QUEUE_STEP&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/precision-attributes/\langle d\rangle, \langle d\rangle</td>
</tr>
<tr>
<td>Exceptions</td>
<td></td>
<td></td>
<td></td>
<td>See Errors on page 18.</td>
</tr>
<tr>
<td>Example</td>
<td></td>
<td>POST</td>
<td></td>
<td>POST <a href="https://Web01:8085/resourcemanagement/rest/resources/precision-queue-steps">https://Web01:8085/resourcemanagement/rest/resources/precision-queue-steps</a></td>
</tr>
</tbody>
</table>

### Precision Queue Resource Item

**Description**

The PrecisionQueue resource defines a queue used for attribute based routing.

A Precision Queue cannot exist on its own: it is always associated with at least one Precision Queue Step. When you create a Precision Queue, you also have to specify the details at least one Precision Queue Step, and the membership between the Precision Queue Step and at least one Precision Attribute. When the Precision Queue...
is created, the system also creates the specified Precision Queue Step or steps and the associated membership items. When you delete a Precision Queue, you must also identify and specify the associated Precision Queue Steps. The system will delete these steps and the associated memberships.

Creating Precision Queues

When you create a Precision Queue, you must specify a composite object that includes:

- the Precision Queue,
- at least one Precision Queue Step,
- for each Precision Queue Step that you specify, at least one Precision Queue Step Precision Attribute Member that links the Precision Queue Step to an existing Precision Attribute.

Optionally, you can specify more than one Precision Queue Step and one or more Precision Queue Step Precision Attribute Members for each Precision Queue Step.

The composite object is passed to the `Create()` method as an ordered array. The following table shows the Precision Queue resource objects required by `Create()`.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Resource Id</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PrecisionQueue</td>
<td>-1</td>
<td>The precision queue to be created.</td>
</tr>
<tr>
<td>PrecisionQueueStep</td>
<td>-1</td>
<td>The first precision queue step to be created and associated with this queue.</td>
</tr>
<tr>
<td>PrecisionQueueStepPrecisionAttributeMember</td>
<td>-3</td>
<td>The resource membership that links this precision queue step to the first precision attribute in the step. The -3 resource ID indicates to the server that this should be created at the same as the parent queue step.</td>
</tr>
<tr>
<td>(PrecisionQueueStepPrecisionAttributeMember)</td>
<td>-3</td>
<td>(Optional) Up to nine additional Precision Queue Step Precision Attribute Members that link the Precision Queue Step to any remaining attributes in the step.</td>
</tr>
<tr>
<td>(Additional Precision Queue Steps and Precision Queue Step precision Attribute Members)</td>
<td>Various</td>
<td>(Optional) Up to nine additional Precision Queue Steps, each followed by between one and ten Precision Queue Step Precision Attribute Members that link the Precision Queue Step to the attributes in the step.</td>
</tr>
</tbody>
</table>

When the Precision Queue resource is created, the following items are also created:

- the Precision Queue Steps that were specified
- for each Precision Queue Step, the corresponding Precision Queue Step Precision Queue Member that links the Precision Queue Step to the Precision Queue
the Precision Queue Step Precision Attribute Members that were specified.

The `Create()` method returns a resource array containing includes the Precision Queue and the associated child resources with their attributes. The array elements are presented in the same order as they were specified in the call to the `Create()` method. Note that Precision Queue Step Precision Queue Member items are created as required, but are not returned in the resource array. The following table shows the Precision Queue resource objects returned by `Create()`.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Resource Id</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PrecisionQueue</td>
<td>Int</td>
<td>The Precision Queue that was created.</td>
</tr>
<tr>
<td>PrecisionQueueStep</td>
<td>Int</td>
<td>The first Precision Queue Step in the precision queue.</td>
</tr>
<tr>
<td>PrecisionQueueStep</td>
<td>-3</td>
<td>The resource membership that links this precision queue step to the first precision attribute in the step. The -3 resource ID indicates to the server that this should be created at the same as the parent queue step.</td>
</tr>
</tbody>
</table>

**Deleting Precision Queues**

When you delete a Precision Queue you must delete a composite object that includes all the Precision Queue Steps as well as the Precision Queue itself.

The composite object is passed to the `Delete()` method as an ordered array. The `Delete()` method deletes the specified Precision Queue Steps and the Precision Queue resource. All associated Precision Queue Step Precision Attribute Members and Precision Queue Step Precision Queue Members are also deleted. The following table shows the resource objects required by `Delete()`.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Resource Id</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PrecisionQueueStep</td>
<td>&lt;id&gt;</td>
<td>The precision queue step(s) to be deleted.</td>
</tr>
<tr>
<td>PrecisionQueue</td>
<td>&lt;id&gt;</td>
<td>The precision queue to be deleted.</td>
</tr>
</tbody>
</table>

**Deleting a Precision Queue: Example**

We recommend using the following steps to delete a Precision Queue resource.

1. Select the Precision Queue to be deleted and find its resource key.
2. Find the Precision Queue Steps associated with the Precision Queue using the search query:

```
childof: "Precision Queue", <resourceId> type: "Precision Queue Step" -status: D latest: 1
```
3. Add the resource keys of the returned Precision Queue Steps to the resource array.

4. Add the Precision Queue resource key to the end of the resource array.

5. Call the `Delete()` method, passing the resource array.

### Adding Precision Queue Steps to Precision Queues

To add a Precision Queue Step to a Precision Queue, create a new Precision Queue Step and specify the id of the Precision Queue in the `PrecisionQueueUrn` field. The member item that links the Precision Queue Step to the required Precision Queue will be created automatically when the Precision Queue Step is created. A Precision Queue can have up to 10 associated Precision Queue Steps.

### Deleting Precision Queue Steps from Precision Queues

To delete a Precision Queue Step from a Precision Queue, specify the id of the required Precision Queue Step.

The member item that links the step to the queue will be deleted automatically. All member items linking attributes to the step will also be deleted automatically.

You cannot delete the last Precision Queue Step in a Precision Queue Step unless you delete the Precision Queue itself.

### Fields

The following fields can be read or set for Precision Queue resources.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AgentOrdering</td>
<td>Int (n-v pair)</td>
<td>Determines the order of agents in a precision queue sub-queue using the following values: 1 = LAA (agent availability time), 2 = Most skilled agent, 3 = Least skilled agent</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>CallOrdering</td>
<td>Int (n-v pair)</td>
<td>Determines the order of calls in this precision queue using the following value: 1 = Priority, then time in queue</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>ServiceLevelThreshold</td>
<td>Int (n-v pair)</td>
<td>The service level threshold in seconds for this precision queue</td>
<td>Yes</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

#### Element Name | Data Type | Description | Required? | Pkey?
--- | --- | --- | --- | ---
ServiceLevelType | Short (n-v pair) | Determines how to calculate the service level for the precision queue using the following values: 1 = ignore abandoned calls, 2 = abandoned call has negative impact, 3 = abandoned call has positive impact | Yes | ✓
ForcedExpandingQueue | Bool (n-v pair) | For future use | No | ✓
DepartmentBizUrn | Int (n-v pair) | Foreign key to the department that this resource is associated with, or -1 if there is no associated department. | No |

---

**Model**

![Precision Queue Model Diagram]

Figure 3-23  Precision Queue Relationships

**REST Protocol**

**Precision Queue Resource REST API Summary**
## Precision Queue Resource REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/precision-queues/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromDate&gt;&amp;toDate=&lt;toDate&gt;…</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/precision-queues</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/precision-queues/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/precision-queue</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationId&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/precision-queues/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/precision-queues</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type@T_PRECISION_QUEUE_MAX@10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/precision-queues/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

**Exceptions**
See Errors on page 18.

**Example**
POST https://Web01:8085/resourcemanagement/rest/resources/precision-queues

---

### Route Resource Item

**Description**

The Route resource represents any possible destination for a call.

**Fields**

The following fields can be read or set for Route resources.
Inherits From: DimensionItem

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skill Group Urn</td>
<td>String</td>
<td>The parent skill group for this route.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Model

**Figure 3-24 Route Relationships**

REST Protocol

**Route Resource REST API Summary**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/routes/&lt;id&gt;/audit?fromDate=&lt;fromDate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
</tbody>
</table>
Route Resource REST API Summary

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted [Base URL]/routes</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted [Base URL]/routes/&lt;id&gt;, &lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK [Base URL]/meta/route</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK [Base URL]/&lt;destinationId&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK [Base URL]/routes/&lt;id&gt;, &lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK [Base URL]/members/routes</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK [Base URL]?queryString=type%3aIT_ROUTE&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted [Base URL]/routes/&lt;id&gt;, &lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/routes

Route Partition Resource Item

Description
The RoutePartition is a collection of route patterns that facilitate call routing by dividing the route plan into logical subsets that are based on organization, location, and call type.

Fields
The following fields can be read or set for Route Partition resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From</td>
<td>DimensionItem</td>
<td>No further fields.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Model

Figure 3-25  Route Partition Relationships

REST Protocol

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/route-partitions/&lt;id&gt;,&lt;id&gt;/audits?fromDate=&lt;fromDate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/route-partitions</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/route-partitions/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/route-partition</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destination id&gt;</td>
</tr>
</tbody>
</table>
### Route Partition Resource REST API Summary

<table>
<thead>
<tr>
<th>Retrieve</th>
<th>Sync.</th>
<th>GET</th>
<th>200 OK</th>
<th>[Base URL]/route-partitions/&lt;id&gt;,&lt;id&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/route-partitions</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_ROUTE_PARTITION&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/route-partitions/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

#### Exceptions
See Errors on page 18.

#### Example
POST https://Web01:8085/resourcemangement/rest/resources/route-partitions

---

### Service Resource Item

**Description**

The Service resource represents a service on a peripheral.

**Fields**

The following fields can be read or set for Service resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inherits From:</strong></td>
<td>DimensionItem</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peripheral Urn</td>
<td>String(10) (n-v pair)</td>
<td>The peripheral or equipment identifier on which the agent will be located. If -1 is supplied then Unified CCMP will pick the Peripheral automatically. Note: cannot be updated after the Create.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>MediaRoutingDomainUrn</td>
<td>String(10) (n-v pair)</td>
<td>The media routing domain that services this service.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Peripheral Number</td>
<td>Int (n-v pair)</td>
<td>This is the number for the service as known to the peripheral it is associated with.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Element Name</td>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
<td>Pkey?</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td>Peripheral Name</td>
<td>String</td>
<td>This is the name of the service as known to the peripheral it is associated with.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Peripheral Service Level Type</td>
<td>Short</td>
<td>This is the type of service level used for the service on the peripheral.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Service Level Type</td>
<td>Short</td>
<td>This is the type that defines how service levels are calculated for the service. This take the values 0: Default; 1: Ignore abandoned calls; 2: Abandoned calls have negative impact; 3: Abandoned calls have positive impact.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Service Level Threshold</td>
<td>Int</td>
<td>This is the threshold in seconds for the service.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Extension</td>
<td>String</td>
<td>This is the extension number for the skill group.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>User Deletable</td>
<td>Bool</td>
<td>Indicates if the item can be deleted (used for parent/child configurations)</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>Department Biz Urn</td>
<td>Int</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
Model

Figure 3-26  Service Relationships

REST Protocol

Service Resource REST API Summary

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/services/&lt;id&gt;/audits?fromDate=&lt;fromdate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/services</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/services/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/services</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationId&gt;</td>
</tr>
</tbody>
</table>
Chapter 3: Resource Management Web Service

Provisionable Resource Types

Service Resource REST API Summary

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Method</th>
<th>Status Code</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/services/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/services</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_SERVICE&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/services/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/services

Skill Group Resource Item

Description
The Skill Group resource represents a queue point in the contact routing solution to which agents are added for the purpose of dealing with the work items, for example, inbound voice calls, emails tasks etc.

Fields
The following fields may be read or set on skill group resource objects.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peripheral Urn</td>
<td>String(10)</td>
<td>The peripheral or equipment identifier on which the skill group will be located. If -l -s supplied then Unified CCMP will pick the Peripheral automatically. Note: cannot be updated after the Create.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>MediaRoutingDomainUrn</td>
<td>String(10)</td>
<td>The media routing domain to be used by this skillgroup. Note: cannot be modified using the Update call.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Peripheral Number</td>
<td>String(50)</td>
<td>This is the number for the skillgroup as known to the peripheral it is associated with.</td>
<td>No</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peripheral Name</td>
<td>String(50)</td>
<td>This is the name for the skillgroup as known to the peripheral it is associated with</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>AvailableHoldoffDelay</td>
<td>Short</td>
<td>This is the number of seconds before an agent becomes available after a call is terminated. Defaults to 0.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Priority</td>
<td>Short</td>
<td>This is the routing priority for the skillgroup. This can take the values 1: Primary; 2: Secondary; 3: Tertiary</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Extension</td>
<td>String(50)</td>
<td>This is the extension number for the service</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>IPTA</td>
<td>Char</td>
<td>This is a character flag (Y or N) indicating whether for this skillgroup the routing platform picks the agent. Defaults to N. Note: Only supported on Unified CCE.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>ServiceLevelThreshold</td>
<td>Int</td>
<td>This is the service level threshold for the skillgroup. Defaults to -1 meaning that if not set the value defaults to that set by the peripheral.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>ServiceLevelType</td>
<td>Short</td>
<td>This is the type that defines how service levels are calculated for the skillgroup. Defaults to 0. This takes the values: 0: Default; 1: Ignore abandoned calls; 2: Abandoned calls have negative impact; 3: Abandoned calls have positive impact.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DefaultEntry</td>
<td>String(50)</td>
<td>This is the default entry for the skillgroup</td>
<td>No</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Provisionable Resource Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubSkillGroupMask</td>
<td>String(64)</td>
<td>A series of characters (Y and N) indicating which sub-skillgroups to create for the skillgroup.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>UserDeletable</td>
<td>Bool</td>
<td>Indicates if the item can be deleted by end users as opposed to the Unified CCMP background processed. Note: used only for Unified CCE parent/child configurations only</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>IsPRSkill</td>
<td>Boolean</td>
<td>(Read only). Indicates whether this skill group is a precision routing skill group.</td>
<td>No</td>
<td>✓</td>
</tr>
<tr>
<td>DepartmentBizUrn</td>
<td>Int</td>
<td>Foreign key to the department that this resource is associated with, or -1 if there is no associated department.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
Model

**Figure 3-27 Skill Group Relationships**

REST Protocol

<table>
<thead>
<tr>
<th>Skill Group Resource REST API Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base URL</strong></td>
</tr>
<tr>
<td><strong>IO Format</strong></td>
</tr>
<tr>
<td><strong>Operation</strong></td>
</tr>
<tr>
<td>Audit</td>
</tr>
<tr>
<td>Create</td>
</tr>
<tr>
<td>Delete</td>
</tr>
<tr>
<td>Describe</td>
</tr>
</tbody>
</table>
Skill Group Resource REST API Summary

<table>
<thead>
<tr>
<th>Method</th>
<th>Sync.</th>
<th>Description</th>
<th>Status</th>
<th>Request URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationid&gt;</td>
</tr>
<tr>
<td>Retrieve</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/skillgroups/&lt;id&gt;,&lt;id&gt;</td>
</tr>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST</td>
<td>200 OK</td>
<td>[Base URL]/members/skillgroups</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]?queryString=type%3aIT_SKILLGROUP&amp;max%3a10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT</td>
<td>202 Accepted</td>
<td>[Base URL]/skillgroups/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

Exceptions
See Errors on page 18.

Example
POST https://Web01:8085/resourcemanagement/rest/resources/skillgroups

Tenant Resource Item

Description
The Tenant resource represents a company or organization. A tenant may exist on dedicated equipment, for example, a dedicated physical or virtualized stack, or may share a portion of an equipment stack. The key definition of a tenant is that it must have telephony/CTI isolation from other tenants on the same equipment stack. For example, in a Cisco UCCE environment, the use of Calling Search Spaces and Route Partitions enforce that Tenant A extensions can't be reached by Tenant B.

Fields
Fields that may be set or read on a Tenant resource item are:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From:</td>
<td>DimensionItem</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No further fields.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Model

Figure 3-28  Tenant Relationships

REST Protocol

<table>
<thead>
<tr>
<th>Tenant Resource REST API Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base URL</strong></td>
</tr>
<tr>
<td><strong>IO Format</strong></td>
</tr>
<tr>
<td><strong>Operation</strong></td>
</tr>
<tr>
<td>Audit</td>
</tr>
<tr>
<td>Create</td>
</tr>
<tr>
<td>Delete</td>
</tr>
<tr>
<td>Describe</td>
</tr>
<tr>
<td>Move</td>
</tr>
<tr>
<td>Retrieve</td>
</tr>
<tr>
<td>Save</td>
</tr>
<tr>
<td>Search</td>
</tr>
<tr>
<td>Update</td>
</tr>
</tbody>
</table>
Tenant Resource REST API Summary

| Exceptions | See Errors on page 18. |
| Example | POST https://Web01:8085/ResourceManagement/rest/resources/tenants |

User Variable Partition Resource Item

Description

The User Variable resource is a user defined variable that can be used in the call routing process.

Fields

The following fields can be read or set for User Variable resources:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionItem</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ObjectType</td>
<td>Guid (n-v pair)</td>
<td>This is a number that indicates the object associated with the user variable. These objects can be skill groups, services etc. The values correspond to the values in OBJECT_TYPE_CODE in TB_DIM_OBJECT_TYPE.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Data Type</td>
<td>Short (n-v pair)</td>
<td>This a number indicating the data type for the user variable. This can take values 0: Long; 1: Float; 2: Char; 3: Date.</td>
<td>Yes</td>
<td>✓</td>
</tr>
<tr>
<td>Persistent</td>
<td>Char (n-v pair)</td>
<td>This is a character flag indicating whether to preserve the value of the user variable between script invocations. Value Y indicates yes.</td>
<td>Yes</td>
<td>✓</td>
</tr>
</tbody>
</table>
REST Protocol

<table>
<thead>
<tr>
<th>Operation</th>
<th>Mode</th>
<th>Method</th>
<th>Status</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/user-variables/&lt;d&gt;/audit?fromDate=&lt;fromDate&gt;&amp;toDate=&lt;toDate&gt;...</td>
</tr>
<tr>
<td>Create</td>
<td>Async.</td>
<td>POST</td>
<td>202 Accepted</td>
<td>[Base URL]/user-variables</td>
</tr>
<tr>
<td>Delete</td>
<td>Async.</td>
<td>DELETE</td>
<td>202 Accepted</td>
<td>[Base URL]/user-variables/&lt;d&gt;</td>
</tr>
<tr>
<td>Describe</td>
<td>Sync.</td>
<td>GET</td>
<td>200 OK</td>
<td>[Base URL]/meta/user-variable</td>
</tr>
<tr>
<td>Move</td>
<td>Sync.</td>
<td>PUT</td>
<td>200 OK</td>
<td>[Base URL]/&lt;destinationId&gt;</td>
</tr>
</tbody>
</table>
### User Variable Resource REST API Summary

<table>
<thead>
<tr>
<th>Retrieve</th>
<th>Sync.</th>
<th>GET 200 OK</th>
<th>[Base URL]/user-variables/&lt;id&gt;,&lt;id&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save</td>
<td>Sync.</td>
<td>POST 200 OK</td>
<td>[Base URL]/members/user-variables</td>
</tr>
<tr>
<td>Search</td>
<td>Sync.</td>
<td>GET 200 OK</td>
<td>[Base URL]?queryString=type%3AIT_USER_VARIABLE&amp;max%3A10</td>
</tr>
<tr>
<td>Update</td>
<td>Async.</td>
<td>PUT 202 Accepted</td>
<td>[Base URL]/user-variables/&lt;id&gt;,&lt;id&gt;</td>
</tr>
</tbody>
</table>

#### Exceptions
See Errors on page 18.

#### Example
POST https://Web01:8085/resourcemanagement/rest/resources/user-variables

#### Provisionable Member Types

##### Member Types, Operations and Relationships

Member types define memberships between entities.

This table lists the supported member types, the operations supported by each member type, and the parent and child items which define each member type.

<table>
<thead>
<tr>
<th>Resource Member Type</th>
<th>Create</th>
<th>Edit</th>
<th>Delete</th>
<th>Read</th>
<th>Parent</th>
<th>Child</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Agent Desktop Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>Agent Desktop</td>
<td>Agent</td>
</tr>
<tr>
<td>Agent Agent Team Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>Agent Team</td>
<td>Agent</td>
</tr>
<tr>
<td>Agent Desktop Dialed Number Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Dialed Number</td>
<td>Agent Desktop</td>
</tr>
<tr>
<td>Agent Peripheral Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>Peripheral</td>
<td>Agent</td>
</tr>
<tr>
<td>Agent Person Member</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>Person</td>
<td>Agent</td>
</tr>
<tr>
<td>Agent Precision Attribute Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Precision Attribute</td>
<td>Agent</td>
</tr>
<tr>
<td>Agent Team Dialed Number Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Dialed Number</td>
<td>Agent Team</td>
</tr>
<tr>
<td>Agent Skill Group Member</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>Skill Group</td>
<td>Agent</td>
</tr>
</tbody>
</table>

Table 3-5   Member Types
<table>
<thead>
<tr>
<th>Resource Member Type</th>
<th>Create</th>
<th>Edit</th>
<th>Delete</th>
<th>Read</th>
<th>Parent</th>
<th>Child</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call Type Routing Script Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Routing Script</td>
<td>Call Type</td>
</tr>
<tr>
<td>Device Profile Directory Number Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Directory Number</td>
<td>Device Profile</td>
</tr>
<tr>
<td>Device Profile IP Endpoint Button Template Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>IP Endpoint Button Template</td>
<td>Device Profile</td>
</tr>
<tr>
<td>Device Profile Person Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Person</td>
<td>Device Profile</td>
</tr>
<tr>
<td>Dialed Number Call Type Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Call Type</td>
<td>Dialed Number</td>
</tr>
<tr>
<td>Dialed Number Routing Client Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>Dialed Number Media Routing Domain Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>Group Group Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Group</td>
<td>Group</td>
</tr>
<tr>
<td>IP Endpoint Button Template Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>IP Endpoint Button Template</td>
<td>IP Endpoint</td>
</tr>
<tr>
<td>IP Endpoint Calling Search Space Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Calling Search Space</td>
<td>IP Endpoint</td>
</tr>
<tr>
<td>IP Endpoint Device Pool Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>Device Pool</td>
<td>IP Endpoint</td>
</tr>
<tr>
<td>IP Endpoint Directory Number Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>Directory Number</td>
<td>IP Endpoint</td>
</tr>
<tr>
<td>IP Endpoint Peripheral Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>Peripheral</td>
<td>IP Endpoint</td>
</tr>
<tr>
<td>Label Dialed Number Member</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Dialed Number</td>
<td>Label</td>
</tr>
<tr>
<td>Label Routing Client Member</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>True</td>
<td>Routing Client</td>
<td>Label</td>
</tr>
</tbody>
</table>
### Member Type Fields

**Fields Common to All Member Types**

All member type classes inherit from the `DimensionMember` class or the `SystemMember` class. The member types in this section also have additional fields.
Agent Agent Team Member Fields

The Agent Agent Team Member class has the following fields:

<table>
<thead>
<tr>
<th>Table 3-6</th>
<th>Agent Agent Team Member Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Element Name</td>
<td>Data Type</td>
</tr>
<tr>
<td>Inherit From: DimensionMember</td>
<td></td>
</tr>
<tr>
<td>Supervisor</td>
<td>Bool (n-v pair)</td>
</tr>
<tr>
<td>PrimarySupervisor</td>
<td>Bool (n-v pair)</td>
</tr>
<tr>
<td>PhysicalMember</td>
<td>Bool (n-v pair)</td>
</tr>
</tbody>
</table>

At least one of Supervisor, PrimarySupervisor and PhysicalMember must be set.

Agent Precision Attribute Member

The Agent Precision Attribute Member class has the following fields:

<table>
<thead>
<tr>
<th>Table 3-7</th>
<th>Agent Precision Attribute Member Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Element Name</td>
<td>Data Type</td>
</tr>
<tr>
<td>Inherit From: DimensionMember</td>
<td></td>
</tr>
<tr>
<td>AttributeValue</td>
<td>String(255) (n-v pair)</td>
</tr>
<tr>
<td>Description</td>
<td>String(255) (n-v pair)</td>
</tr>
</tbody>
</table>
Agent Skillgroup Member

The AgentSkillgroupMember class has the following fields:

**Table 3-8 Agent Skillgroup Member Fields**

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefaultMember</td>
<td>Bool (n-v pair)</td>
<td>Indicates if this is the default skillgroup for the agent.</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Device Profile Directory Number Member

The DeviceProfileDirectoryNumberMember class has the following fields:

**Table 3-9 Device Profile Directory Number Member Fields**

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>LinelIndex</td>
<td>Bool (n-v pair)</td>
<td>This is a 0 based index for the Directory Numbers associated with this Device Profile. This must be unique for a particular Device Profile.</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>MaxNumCalls</td>
<td>Int (n-v pair)</td>
<td>The maximum number of calls to the directory number.</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>BusyTrigger</td>
<td>Int (n-v pair)</td>
<td>The configuration per line appearance and per cluster for a Directory number after which the call to that specific Directory Number is rejected with a cause of busy.</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Provisionable Member Types

#### Element Name | Data Type | Description | Required? | Pkey?
--- | --- | --- | --- | ---
RingSetting | String(100) (n-v pair) | A string representing the ring setting. One of the following: Disable, Flash Only, Ring Once, Ring, Beep Only | No | Yes
TemplateData | Xml (n-v pair) | The template data to be used. | No | Yes

#### Dialed Number Call Type Member

The `DialedNumberCallTypeMember` class has the following fields:

**Table 3-10 Dialed Number Call Type Member Fields**

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionMember</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RegionBizUrn</td>
<td>Int (n-v pair)</td>
<td>The URN as in TB_DIM_REGION.</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Item</td>
<td>Int (n-v pair)</td>
<td>This is a sequence in which rows for a dialed number are tested against the call qualifiers.</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>CLIWildCard</td>
<td>Int (n-v pair)</td>
<td>A region name or a prefix indicating the leading digits of a telephone number. This can also be a complete telephone number.</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Provisionable Member Types

**Table 3-11  Provisionable Member Types**

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLI WildCardType</td>
<td>Int</td>
<td>The type of the CLI wild card. One of:</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>0 Unknown</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 NPA(3 digit match)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 NPA-NXX(6 digit match)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3 Match(all digit match)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4 Region</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>5 All (match all CLIs)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6 Prefix</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CEDWildCard</td>
<td>String(60)</td>
<td>A wild card value to be matched with the CED. One of:</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>_A All</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>_NR Not required</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>_NE Not entered</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>_N None required or entered</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>String(255)</td>
<td>A string containing any additional information about the</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td>membership.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**IP Endpoint Directory Number Member**

The **IP Endpoint Directory Number Member** class has the following fields:

*Table 3-11  IP Endpoint Directory Number Member Fields*

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From: DimensionMember</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LinelIndex</td>
<td>Int</td>
<td>This is a 0 based index for the Directory Number associated with the IP Endpoint (Phone). This must be unique for a particular IP Endpoint.</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(n-v pair)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Provisionable Member Types

#### MaxNumCalls
- **Element Name:** MaxNumCalls
- **Data Type:** Int (n-v pair)
- **Description:** The maximum number of calls to the directory number. The default is 2.
- **Required:** No
- **Pkey:** Yes

#### BusyTrigger
- **Element Name:** BusyTrigger
- **Data Type:** Int (n-v pair)
- **Description:** The configuration per line appearance and per cluster for a Directory number after which the call to that specific Directory Number is rejected with a cause of busy. The default is 1.
- **Required:** No
- **Pkey:** Yes

#### TemplateData
- **Element Name:** TemplateData
- **Data Type:** Xml (n-v pair)
- **Description:** The template data.
- **Required:** No
- **Pkey:** Yes

### Precision Queue Step Precision Attribute Member Fields

The `PrecisionQueueStepPrecisionAttributeMember` class has the following fields:

#### Table 3-12 Precision Queue Step Precision Attribute Member Fields

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherit From:</td>
<td>DimensionMember</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value1</td>
<td>String(255) (n-v pair)</td>
<td>The value that the attribute is tested against. It must be able to be converted to the <code>AttributeDataType</code> specified for the related Precision Attribute.</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>TermOrder</td>
<td>Int (n-v pair)</td>
<td>The order of the terms in a Precision Queue Step. This value must start at 1 (zero is invalid) for each new Precision Queue Step and increment by 1 for each subsequent term in the Precision Queue Step.</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
### Table 3-13 Skillgroup Service Member Fields

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From:</td>
<td></td>
<td>Di mensionMember</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Skillgroup Service Member**

The `SkillgroupServiceMember` class has the following fields:

```
Table 3-13 Skillgroup Service Member Fields

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inherits From:</td>
<td></td>
<td>Di mensionMember</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Provisionable Member Types

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
<th>Pkey?</th>
</tr>
</thead>
<tbody>
<tr>
<td>PriorityLevel</td>
<td>Int (n-v pair)</td>
<td>This is the priority level of the specified service for the specified skill group.</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Creating New Member Types

To create a new member, pass a new resource object to the Create() method with the type set to the appropriate ResourceMemberType and with the parent and child ids set accordingly.

**Note**

Precision Queue Step Precision Queue Members cannot be created explicitly using Create(). They are always created automatically when the corresponding Precision Queue Step is created. This is because each Precision Queue Step must always be associated with a Precision Queue: a Precision Queue Step cannot exist separately from a Precision Queue.

Member Pkey Types

<table>
<thead>
<tr>
<th>Pkey Member Type</th>
<th>Create</th>
<th>Edit</th>
<th>Delete</th>
<th>Read</th>
<th>Parent</th>
<th>Child</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Agent Team Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Agent</td>
<td>Agent Team</td>
</tr>
<tr>
<td>Agent Peripheral Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Peripheral</td>
<td>Agent</td>
</tr>
<tr>
<td>Agent Precision Attribute Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Precision Attribute</td>
<td>Agent</td>
</tr>
<tr>
<td>Agent Skill Group Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Skill Group</td>
<td>Agent</td>
</tr>
<tr>
<td>Call Type Routing Script Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Routing Script</td>
<td>Call Type</td>
</tr>
<tr>
<td>Device Profile Directory Number Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Directory Number</td>
<td>Device Profile</td>
</tr>
<tr>
<td>Dialed Number Call Type Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Call Type</td>
<td>Dialed Number</td>
</tr>
<tr>
<td>IP Endpoint Directory Number Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Directory Number</td>
<td>IP Endpoint</td>
</tr>
<tr>
<td>Precision Queue Step Precision Attribute Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Precision Attribute</td>
<td>Precision Queue</td>
</tr>
</tbody>
</table>

Web Services Reference for Cisco Unified Contact Center Management Portal

August 2015

130
### Pkey Member Type

<table>
<thead>
<tr>
<th>Pkey Member Type</th>
<th>Create</th>
<th>Edit</th>
<th>Delete</th>
<th>Read</th>
<th>Parent</th>
<th>Child</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query Rule Campaign Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Campaign</td>
<td>Query Rule</td>
</tr>
<tr>
<td>Route Partition Calling Search Space Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Calling Search Space</td>
<td>Route Partition</td>
</tr>
<tr>
<td>Skill Group Campaign Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Campaign</td>
<td>Skill Group</td>
</tr>
<tr>
<td>Skill Group Service Member Pkey</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>Service</td>
<td>Skill Group</td>
</tr>
</tbody>
</table>

### Web Service APIs

#### Create

**Description**

The Create() API creates a supported provisionable item type. For example, use this API to create an Agent.

In general the provisioning operation is synchronous for system resources such as users and folders and asynchronous for remote resources such as agents. Remote resources are queued for provisioning onto the underlying equipment, based on the effective from date field. They are provisioned when they become effective and when there is bandwidth on the remote equipment.

**Parameters**

The Create() API expects the following parameters:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource[]</td>
<td>An array of resources to be provisioned to the Contact Center environment (see section Resource Hierarchy Description).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Return Type**

The Create() API returns the following objects:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestResult[]</td>
<td>An array of results from the resources to be created (see section Request Result).</td>
<td>Yes</td>
</tr>
</tbody>
</table>
REST Protocol

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td><code>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/&lt;resourceType&gt;s</code></td>
</tr>
<tr>
<td>HTTP Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><code>https://APPSRV01:8085/ResourceManagement/rest/Resources/folders</code></td>
</tr>
</tbody>
</table>

Sequence Diagram

*Figure 3-30  Sequence Diagram for Resource Management Create API*
Creating Items with Pkey Maps

When you create a resource or membership that is mapped to more than one equipment instance, you need to specify a pkey map in EquipmentMapping for each equipment instance. To do this:

- In Fields set the item fields that are not equipment-specific, or are common to all equipment instances. These values will be applied to all equipment instances.
- In EquipmentMapping create a pkey map resource for each equipment instance (identified by a unique value for ClusterResourceId) and populate it with the fields that have equipment-specific values. These values will only apply to the specified equipment instance and, if necessary, will override any common values that were specified in Fields.

For example, for an Agent on multiple equipment instances:

- `AgentDesktopUrn` and `PersonUrn` are not specific to any equipment, so will be always be included in Fields of the Agent resource.
- A specific Agent may be a Supervisor on all the equipment instances, so the `Supervisor` flag can be included in Fields of the Agent resource if required.
- A specific Agent will need a `AgentPkey` resource in EquipmentMapping for each equipment instance, with ClusterResourceId set to the unique value for that equipment instance.
- A specific Agent may have one name on one equipment instance and a different name on another equipment instance, so `InternalName` can be set separately for each AgentPkey resource in EquipmentMapping if required.

Additionally, if you want to create several similar items on different equipment instances, you can specify a comma-separated list of equipment instances in Fields. EquipmentMapping. This will automatically create a pkey map resource in EquipmentMapping for each item in the list, and populate with ClusterResourceId the corresponding value from the list. The values for the other fields will be as specified in Fields. If required, you can specify a comma-separated list of equipment instances in Fields. EquipmentMapping as well as creating one or more pkey map resources for specific equipment instances with specified ClusterResourceId values in EquipmentMapping.

For example, you may create an Agent with the same details on two equipment instances:

- Set the Fields of the Agent resource to the common values to be applied to both equipment instances.
- Set Fields. EquipmentMapping to the unique identifiers for the two equipment instances, separated by a comma.
Update

Description
The Update() API updates supported provisionable item types. For example, use this API to add a Skill Group membership to an Agent.

Parameters
The Update() API expects the following parameters:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource[]</td>
<td>An array of resources to be provisioned to the Contact Center environment (see section Resource Hierarchy Description).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Return Type
The Update() API returns the following objects:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestResult[]</td>
<td>An array of result from the resources to be updated (see section Request Result).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

REST Protocol

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><a href="https://APPSRV01:8085/ResourceManagement/rest/Resources/agent/s/315553">https://APPSRV01:8085/ResourceManagement/rest/Resources/agent/s/315553</a></td>
</tr>
</tbody>
</table>
Sequence Diagram

Figure 3-31  Sequence Diagram for Resource Management Update API

Updating Items with Pkey Maps

When you update a resource or membership that is mapped to more than one equipment instance, to access equipment-specific information, you need to specify a pkey map in EquipmentMapping for the equipment instance.

Specify the pkey map in the same way as described in section Creating Items with Pkey Maps on page 133.

You can use Update() to add a new pkey map, but you cannot use Update() to delete an existing pkey map. To delete an existing pkey map from an item, use Delete() on the corresponding pkey item.
Delete

Description

The `Delete()` API deletes supported provisionable item types. For example, use this API to delete a Skill Group.

Parameters

The `Delete()` API expects the following parameters:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceKey[]</td>
<td>An array of resource identities for the items to be deleted (see section Resource Hierarchy Description).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Return Type

The `Delete()` API returns the following objects:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestResult[]</td>
<td>An array of results from the resources to be deleted (see section Request Result).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

REST Protocol

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
</tbody>
</table>
Sequence Diagram

Figure 3-32 Sequence Diagram for Resource Management Delete API

Deleting Items with Pkey Maps

When you delete a resource or membership that is mapped to more than one equipment instance, all pkey maps are deleted with the item.

If you want to delete a resource or membership mapping to a specific equipment instance without deleting the item itself, use the Update() method on the parent item, and set the field status to deleted.

You cannot delete the last pkey map for a resource or membership. You must delete the item itself.

Note
Save

Description

The Save() API allows you to create, update and delete multiple item and types in a single API call.

Use this API

- for efficiency to avoid provisioning separate create and delete operations for the same item, (for example, when reskilling many agents at the start of a shift, which requires a delete followed by an add)
- when two different operations have to be performed concurrently to preserve data integrity (for example, when creating a new Precision Queue Step in a Precision Queue and updating an existing Precision Queue Step in the Precision Queue at the same time).

Parameters

The Save() API expects the following parameters:

<table>
<thead>
<tr>
<th>Table 3-14 Parameters for Save API</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SOAP Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>resourcesToAdd</td>
<td>Resource[]</td>
<td>An array of resources to be created or updated. Must be supplied, but may be empty.</td>
<td>Yes</td>
</tr>
<tr>
<td>resourcesToDelete</td>
<td>Resource[]</td>
<td>An array of resources to be deleted. Must be supplied, but may be empty.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Return Type

The Save() API returns the following objects:

<table>
<thead>
<tr>
<th>Table 3-15 Return Type for Create API</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestResult[]</td>
<td>An array of results from the resources to be created, updated or deleted.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
## REST Protocol

**Table 3-16 REST Protocol URI for Save API**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| URL       | If request contains all the same type of resource:  
            {resourceMemberTypeCollection}  
            If request contains more than one type of resource:  
| HTTP Method | POST |
| Input/Output format | XML or JSON |
| Example | https://APPSRV01:8085/ResourceManagement/rest/Resources/members/folders |
Saving Items with Pkey Maps

When you save a resource or membership that is mapped to more than one equipment instance, the pkey map is handled in the same way as for the underlying create, update or delete operation.
**Move**

**Description**

The `Move()` API moves one or more resource items to the specified folder from their current folder location or locations.

For example, you can use this API to move multiple Agents to a new folder in a single action, instead of using `Update()` and needing to specify the destination folder separately for each Agent.

**Parameters**

The `Move()` API expects the following parameters:

*Table 3-17 Parameters for Move API*

<table>
<thead>
<tr>
<th>SOAP Element Name</th>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>destinationFolderId</td>
<td>Int</td>
<td>The identifier of the folder where the resource items will be moved to.</td>
<td>Yes</td>
</tr>
<tr>
<td>resourceKeys</td>
<td>ResourceKey[]</td>
<td>An array of resource identities for the items to be moved (see Resource Hierarchy Description).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Return Type**

The `Move()` API returns the following objects:

*Table 3-18 Return Type for Move API*

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestResult[]</td>
<td>An array of results from the resources to be moved (see section Request Result).</td>
<td>Yes</td>
</tr>
</tbody>
</table>
## REST Protocol

### Table 3-19  REST Protocol URI for Move API

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td><code>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/destination/{destinationFolderId}</code></td>
</tr>
<tr>
<td>HTTP Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><code>https://APPSRV01:8085/ResourceManagement/rest/Resources/destination/{destinationFolderId}</code></td>
</tr>
</tbody>
</table>
Moving Items with Pkey Maps

When you move a resource that is mapped to more than one equipment instance, the pkey map is handled in the same way as for the underlying update operation.

Retrieve

Description

The Retrieve() API returns a collection of Resource objects that relate to the supplied ResourceKey objects.

Use this API to retrieve collections of specific resources for which information is to be displayed, for example, when a Skill Group is selected from a list of Skill Groups and the properties specific to that Skill Group are to be displayed.
Parameters

The Retrieve() API expects the following parameters:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource[]</td>
<td>An array of resource identities for the items to be retrieved (see section Resource Hierarchy Description).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Return Type

The Retrieve() API returns the following objects:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource[]</td>
<td>An array of resources of the items relating to the identities passed in. Items will be returned in the order of their corresponding identities as passed into the method (see section Resource Hierarchy Description).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

REST Protocol

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input/ Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><a href="https://APPSRV01:8085/ResourceManagement/rest/Resources/agents/315553">https://APPSRV01:8085/ResourceManagement/rest/Resources/agents/315553</a></td>
</tr>
</tbody>
</table>
Retrieving Items with Pkey Maps

When you retrieve a resource or membership that is mapped to more than one equipment instance, the following applies:

- **Fields** contains the details for all fields that are not equipment-specific.
- **Fields** also contains the equipment-specific details for the earliest equipment mapping that is still active.
- **EquipmentMapping** contains a pkey map for each of the other equipment instances. This pkey map only contains the fields that are specific to that equipment instance.

This means that retrieved resources and memberships can be accessed successfully by legacy clients that do not expect pkey map details as well as by clients that are aware of the pkey map resource type.

Search

Description

The Search() API retrieves items based on item type, name, relationships etc. The client can specify search terms to control the search. The client can also specify a collection of items to be excluded from the query results, even if they match the specified search terms.

Parameters

The Search() API expects the following parameters:
### Data Type

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>String</strong></td>
<td>The search string for which Resource items should be retrieved.</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>ResourceKey[]</strong></td>
<td>A collection of ResourceKey objects for items that should be excluded from the search results.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Search String Syntax

The search string is constructed from one or more of search terms. Table 3-20 lists the available search terms, using the following conventions:

- Text enclosed by `{}` indicates an expression to be replaced with your own value, for example `{Id}` must be replaced with an item id.
- An ellipsis (\ldots) indicates that the preceding term or phrase may be repeated as many times as required. For example, the search term syntax 
  \[
  \text{item:}\{\text{IdOrName}\}, \{\text{IdOrName}\}\ldots
  \]
  indicates that the `item` search term requires one value that is an id or name, followed optionally by as many more id or name values as are required, each separated by a comma.
- Double quotes (""") must be used to delimit strings that contain embedded spaces.

Search terms can be combined or negated to give a powerful and flexible mechanism for retrieving specific items.

### Negation

To negate a search term, add a "-" character before the search term. Any items that match the search term following the "-" character will be excluded from the result set.

For example, the search term

\[
\text{folder:/cicm -text:"HIDDEN ITEM"
}
\]
gives a result set containing all items in the /cicm folder except those containing "HIDDEN ITEM" in their name or description.

### Combined Search Expressions

Search terms can be combined using AND and OR operators to build up a search expression. By default, specifying two search terms with a space between the terms is interpreted as an AND operation, and only those results that meet both criteria are returned. OR operators can be applied to expressions but cannot be used to create nested expressions.

For example, the search term

\[
\text{folder:/cicm name:"Bob" OR folder:/cicm2 name:"Dave"
}
\]

will return an item called “Dave” in the /cicm2 folder and an item called ”Bob” in the /cicm folder but will not return an item called” Dave” in the /cicm folder.
Global Search Terms

Global search terms can be defined which apply to the entire search string. Global search terms are marked with a "*" before the search term.

For example, the search string

```
*folder:/cicm name: "Bob" OR name: "Dave"
```

applies the *folder:/cicm search filter to the entire search string, and will return an item "Dave" in the /cicm folder and an item "Bob" in the /cicm folder but will not return an item "Dave" in the /cicm2 folder.

Modifier Search Terms

The following search terms are modifiers, and are used to modify the result set returned by the other search terms. They cannot be used on their own:

- count
- effective
- effective range
- max
- modified
- offset
- owner
- sort

Search Performance

The following search terms are efficient, and can be used to restrict the search results set before applying any specific property-based searches (which are slower):

- deleted
- effective
- effective range
- enabled
- hidden
- latest
- modified
- status
- system
- type

For best search performance, where possible, use the search terms in the list above instead of using a property search on the same field, which will be slower. The following search terms use property searches:

- property
- childof (if {MemberOptions} is specified)
- parentof (if {MemberOptions} is specified).
For example, the search string

```plaintext
del et ed: fals e  type: “Agent  Agent  Team Member”  chi l dof : “Agent  Team”, 1234
```

will be more efficient than

```plaintext
chi l dof : “Agent  Team”, 1234[ Type=“Agent  Agent  Team Member”;  
Del et ed=f al s e]
```

although both search strings will return all non-deleted Agent Agent Team Member items for the Agent Team “1234”.

**Search Latency**

The Search API uses Enterprise Level Caching, which may lead to a delay between Create, Update or Delete operations and their state being reflected in the search results. It is recommend that clients maintain a list of changes that may have not been propagated to the cache at the point the search is called. The SOAP client can use the subscriptions mechanism to subscribe for notifications when items are provisioned, and remove the items from the list as the notifications are received.

**Search Terms**

**Table 3-20  Search Terms for Search API**

<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| count       | Return the number of items that would be returned by the specified search terms. | Syntax  
```plaintext
count:
```

Examples
```plaintext
cstext: "Skill Group For Spanish" count:
```
returns the number of resources that match the cstext search term "Skill Group For Spanish".
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| cstext      | Case-sensitive search for items with the specified text in the Name, Internal Name or Description fields. This search is much slower than the key search but provides a finer grained search. | **Syntax**  
One of:  
cstext: {SearchString}  
cstext: {SearchString}, {SearchField}  
{SearchString} specifies the string to search for, and the optional {SearchField} is one of:  
a search for specified text in name and Internal Name fields only  
n search for specified text in Name field only  
i search for specified text in Internal Name field only  
d search for specified text in Description field only  

**Examples**  
cstext: "Spanish" returns items where Name, Internal Name or Description contains “Spanish” (but not “SPANISH”, or “spanish” etc).  
cstext: "SkillGroup For Spanish".  
cstext: "David",i returns items where the internal name contains “David”, but not items where the internal name contains “david” or “DAVID".
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| childof     | Restrict the search to children of the specified parent. Both a type and an id are needed to ensure the parent is identified uniquely. If the type is not supplied, then the children of all items with the specified id will be returned. If the id is not supplied then the children of all items of the specified type will be returned. Where possible, specify both a type and an id as this makes the search more efficient. | Syntax
One of:
childof: {Type}
childof: {Id}
childof: {Type}, {Id}
childof: {Type}, {Id}[{MemberOptions}]
childof: {Id}[{MemberOptions}]
childof: {Type}, {Id}[{MemberOptions}]
The optional {MemberOptions} is enclosed in square brackets and specifies one or more property filters to apply to the membership children of the specified parent. Each membership property filter in {MemberOptions} is of the form {PropertyName}{Comparison}{PropertyValue}, where {Comparison} is one of:
= is equal to 
! is not equal to 
> is greater than 
< is less than 
} is greater than or equal to 
} is less than or equal to
If {MemberOptions} contains more than one membership property filter, each filter must be separated by a semi-colon (;). For example, [Type="Agent Skillgroup Member"; DefaultMember=true], specifies a membership property filter for Agent Skillgroup Members where the membership represents the default skillgroup for that agent. **Note**: The {MemberOptions} property filter only affects child membership items. It does not affect child resource items, and all child resource items of the specified parent will be returned, even with a {MemberOptions} filter specified. If you want to restrict the result set further, include one of the other search terms, for example, type: or property:.

Examples
childof: "Skill Group", 1234 returns all child items associated with the Skill Group with id "1234".
childof: 1234 returns all child items associated with any item with id "1234".
childof: Peripheral returns all child items associated with all Peripherals.
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| deleted     | Specify whether to include deleted items in the search. By default, the search results will include both deleted and non-deleted items. | Syntax  
`deleted: {Flag}`  
{Flag} is one of `false`, `true`, `0` (false) or `1` (true)  
Examples  
`deleted: true` returns only deleted items.  
`deleted: false` returns only non-deleted items.  
`deleted: 0` returns only non-deleted items. |
| effective   | Restrict the search to items that were effective at the specified date and time. | Syntax  
`effective: {DateTime}`  
{DateTime} is specified as `yyyy-mm-ddThh:mm:ss`  
Example  
`effective: "2009-09-20T23:00:00"` returns only items that were effective at 2300 on 20th September 2009. |
| effective range | Restrict the search to items that were effective during the supplied date and time range. | Syntax  
`effectiveRange: {StartDateTime}, {EndDateTime}`  
{StartDateTime} and {EndDateTime} are specified as `yyyy-mm-ddThh:mm:ss`  
Example  
`effectiveRange: "2009-09-20T23:00:00", "2009-09-21T23:00:00"` returns only items that were effective between 2300 on 20th September 2009 and 2300 on 21th September 2009. |
| enabled     | Specify whether to include enabled items in the search. By default, the search results will include both enabled and disabled items. | Syntax  
`enabled: {Flag}`  
{Flag} is one of `false`, `true`, `0` (false) or `1` (true)  
Examples  
`enabled: true` returns only enabled items.  
`enabled: false` returns only disabled items.  
`enabled: 1` returns only enabled items. |
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| folder      | Restrict the search to items in the specified folder, and optionally, subfolders or parent folders. This search filter only returns items that are in the folder tree. It will return remote resources and system resources (including other folders) but not member items. | **Syntax**<br>One of:<br>`folder: {PathOrId}`<br>`folder: {PathOrId}{Options}`<br>The optional `{Options}` is one of:<br>`/*` Search subfolders (one level)<br>`*` Search folder and subfolders (one level)<br>`**` Search folder and subfolders (all levels)<br>`/**` Search subfolders (all levels)<br>`^` Search parent folders (all levels)<br>If `{Options}` is not specified, then the search is restricted to the specified folder.<br><br>**Examples**<br>`folder: /cicm` returns items (including subfolders) in `/cicm`. <br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`<br>`folder: /cicm`
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| hidden      | Specify whether to include hidden items in the search. By default, the search results will include both hidden and non-hidden (visible) items. | **Syntax**  
hidden: {Flag}  
{Flag} is one of false, true, 0 (false) or 1 (true)  
**Examples**  
hidden: true returns only hidden items.  
hidden: 0 returns only visible items. |
| internal name | Search for items by internal Name (which corresponds to the name of the item on the remote equipment, for example, EnterpriseName in Unified CCE). | **Syntax**  
internalname: {Name}  
**Examples**  
internalname: Agent1 returns all resources with internal name “Agent1”.  
internalname: "Agent 1234" returns all resources with internal name of “Agent 1234”. |
| item         | Search for items by id, Name or internal Name. Both a type and an id are needed to ensure the item is identified uniquely. If a type: search term is not specified as well, then all items with the specified id will be returned. | **Syntax**  
One of:  
item {IdOrName}  
item {IdOrName}, {IdOrName}...  
If more than one {IdOrName} is specified, then each one must be separated by a comma (,).  
**Examples**  
item 1234 returns all items with an id, name or internal name of “1234”.  
item 1234, 7654 returns all items with an id, name or internal name of “1234” or “7654”.  
item 1234, 7654 type: Agent returns all Agents with an id, name or internal name of “1234” or “7654”.  
item "1234, Bob1" returns all items with a name or internal name of "1234,Bob1". |
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| key         | Search for items by the item key. This search term accepts a single item or a list of items, which may be of different types. This search is very efficient. | **Syntax**  
One of:  
key: {Type}, {Id}  
key: {Type}, {Id}|{Type}, {Id}...  
If more than one {Type}, {Id} pair is specified, then each {Type}, {Id} pair must be separated by a vertical bar (|).  
**Example**  
key: Agent, 9474|Peripheral, 2917|MT_ITEM_TENANT_MEMBER, 2927 returns the Agent with id 9474, the Peripheral with id 2917 and the MT_ITEM_TENANT_MEMBER with id 2927. |
| latest      | Specify whether to include current items and items with type-2 changes in the search. By default, the search results will include both current items and items with type-2 changes. | **Syntax**  
latest: {Flag}  
{Flag} is one of false, true, 0 (false) or 1 (true)  
**Examples**  
latest: 1 returns only current items.  
latest: false returns only items with type-2 changes. |
| max         | Restrict the number of items returned. Combine this with offset to return paged results. | **Syntax**  
max: {Count}  
**Examples**  
max: 2000 returns only the first 2000 matching items. |
| memberchildof | Search for member objects of the specified type which link to the specified parent.  
**Note.** This search term is included for legacy use only. New code should use the childof: search term to specify the parent, together with the type: search term to specify the required member type. | **Syntax**  
memberchildof: {ParentType}, {ParentId} {MemberType}  
**Example**  
memberchildof: "Agent Team", 4567, "Agent Agent Team Member" returns all Agent Agent Team Member objects which have Agent Team 4567 as the parent. |
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>member parent of</td>
<td>Search for member objects of the specified type which link to the specified child.</td>
<td><strong>Note.</strong> This search term is included for legacy use only. New code should use the <code>parent of:</code> search term to specify the child, together with the <code>type:</code> search term to specify the required member type.</td>
</tr>
<tr>
<td>modified</td>
<td>Restrict the search to items modified on or after the specified date time.</td>
<td><strong>Syntax</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>One of:</td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>modified: {DateTime}</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>modified: {DeltaTime}</code></td>
</tr>
<tr>
<td>name</td>
<td>Search for items by name.</td>
<td><strong>Syntax</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>name: {Name}</code></td>
</tr>
</tbody>
</table>

**Examples**

- `name: Agent1` returns the Agent "Agent1".
- `name: "Agent 1234"` returns the Agent "Agent 1234".

**Example**

- `modified: "2009-09-20 23:00:00"` returns only items that were modified on or after 2300 on 20th September 2009.
- `modified: -1hour` returns only items that were modified within the last hour.
- `modified: -90seconds` returns only items that were modified within the last 90 seconds.
- `modified: -7days` returns only items that were modified within the last week.
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| **owner**   | Restrict the items returned to those owned by the specified tenant. | **Syntax**  
One of:  
owner: {IdOrTenant}  
owner: {IdOrTenant}, {IdOrTenant}...  
If more than one {IdOrTenant} is specified, then each one must be separated by a comma (,).  
**Examples**  
owner: /Tenant1 returns only items owned by Tenant1.  
owner: {24811131-76E9-4406-9F66-711FD8716955}, {2ADA5C58-839E-4EC8-901E-1692E224B132} returns only items owned by the tenants with the specified ids. |
| **offset**  | Restrict the items returned to those after the specified position in the results array. Combine this with max to return paged results.  
If the specified offset is greater than the number of matching results, the results array is empty. | **Syntax**  
offset: {Offset Count}  
**Example**  
offset: 100 max: 50 returns only the 101st to 150th matching items. |
### Search Term

**parent of**

### Description

Restrict the search to parents of the specified child. Both a type and an id are needed to ensure the child is identified uniquely. If the type is not supplied, then the parents of all items with the specified id will be returned. If the id is not supplied then the parents of all items of the specified type will be returned. Where possible, specify both a type and an id as this makes the search more efficient.

### Syntax and Examples

#### Syntax

One of:

- `parent of: {Type}`
- `parent of: {Id}`
- `parent of: {Type},{Id}`
- `parent of: {Type} [MemberOptions]`
- `parent of: {Id} [MemberOptions]`
- `parent of: {Type},{Id} [MemberOptions]`

The optional `{MemberOptions}` is enclosed in square brackets and specifies one or more property filters to apply to the membership parents of the specified child.

Each membership property filter in `{MemberOptions}` is of the form `{PropertyName}{Comparison}{PropertyValue}`, where `{Comparison}` is one of:

- `=` is equal to
- `!=` is not equal to
- `>` is greater than
- `<` is less than
- `}` is greater than or equal to
- `{` is less than or equal to

If `{MemberOptions}` contains more than one membership property filter, each property filter must be separated by a semi-colon (;). For example, `{Type="Agent Skillgroup Member"; DefaultMember=true}`, specifies a membership property filter for Agent Skillgroup Members where the membership represents the default skillgroup for that agent.

**Note:** The `{MemberOptions}` property filter only affects parent membership items. It does not affect parent resource items, and all parent resource items of the specified child will be returned, even with a `{MemberOptions}` filter specified. If you want to restrict the result set further, include one of the other search terms, for example, `type:` or `property:`.

#### Examples

- `parent of: Agent,1234` returns all parent items associated with the Agent with id “1234”.
- `parent of: 1234` returns all parent items associated with any item with the id “1234”.
- `parent of: "Agent"` returns all parent items associated with any Agent.
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| property    | Search for items based on the value of one or more of their properties. This search is case sensitive. | **Syntax**

    property: {PropertyName}{Comparison}{PropertyValue}

{Comparison} is one of:

    =  is equal to
    !  is not equal to
    >  is greater than
    <  is less than
    }  is greater than or equal to
    {  is less than or equal to

**Examples**

    property: FirstName=Bob returns all items where the first name is “Bob”.
    property: LastName!=Smith returns all items where the last name is not “Smith”.
    property: WaitTime>20 returns all items where the wait time is greater than 20 seconds.
    property: Timeout}20 returns all items where the timeout is greater than or equal to 20 seconds.
<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>sort</td>
<td>Sort the returned items by the specified property.</td>
<td><strong>Syntax</strong>&lt;br&gt;One of:&lt;br&gt;<code>sort:</code> <code>{Sort Term}</code>&lt;br&gt;<code>sort:</code> <code>{Sort Term}$</code> <code>{Sort Term}</code>...&lt;br&gt;{Sort Term} is one of:&lt;br&gt;{PropertyName} (an ascending sort on the specified property name)&lt;br&gt;{PropertyName}, ASC (an ascending sort on the specified property name)&lt;br&gt;{PropertyName}, DESC (a descending sort on the specified property name)&lt;br&gt;If more than one {Sort Term} is specified, then each one must be separated by a dollar sign ($). Note that {propertyName} is case-sensitive.&lt;br&gt;&lt;br&gt;<strong>Examples</strong>&lt;br&gt;<code>sort: InternalName</code> returns the matching items, sorted by the Internal Name field in ascending order.&lt;br&gt;<code>sort: InternalName, DESC</code> returns the matching items, sorted by internal name, in descending order.&lt;br&gt;<code>sort: Supervisor$InternalName, DESC</code> returns the matching items, sorted first by supervisor in ascending order, then by internal name field, in reverse order.</td>
</tr>
<tr>
<td>status</td>
<td>Restrict the search to items matching the specified status.</td>
<td><strong>Syntax</strong>&lt;br&gt;<code>status:</code> <code>{Stat us}</code>&lt;br&gt;&lt;br&gt;<strong>Examples</strong>&lt;br&gt;<code>status: R</code> only returns items that are in the ready state.&lt;br&gt;<code>status: D</code> only returns items that are in the deleted state.</td>
</tr>
</tbody>
</table>
### Search Term: `system`

Specify whether to include items that are owned by the system in the search. By default, the search results will include both system items and non-system items. Items that are owned by the system include the root folder, system search folders, and the users, groups and roles that are created when the system is first installed. These items cannot be edited or deleted.

**Syntax**

```plaintext
system {Flag}
```

**Examples**

- `system: true` only returns items owned by the system.
- `system: false` only returns items not owned by the system.
- `system: 0` only returns items not owned by the system.

### Search Term: `text`

Search for items with the specified text in the Name, InternalName or Description fields. This search is not case-sensitive. This search is much slower than the key search but provides a finer grained search.

**Syntax**

One of:

- `text: {SearchString}`
- `text: {SearchString}, {SearchField}`

{SearchString} specifies the string to search for, and the optional {SearchField} is one of:

- `a`: search for specified text in Name and InternalName fields only
- `n`: search for specified text in Name field only
- `i`: search for specified text in InternalName field only
- `d`: search for specified text in Description field only

**Examples**

- `text: "Spanish"` returns items where Name, InternalName or Description contains “Spanish” (or “SPANISH”, or “spanish” etc)
- `text: "Skill Group For Spanish"`
- `text: "David",i` returns items where InternalName contains “David”, and also items where InternalName contains “david” or “DAVID".
## Search Term

<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| type        | Restrict the search to items of the specified type. | **Syntax**
  type: {Type}

**Examples**
  type: IT_AGENT
  type: "Agent Team"
  type: Folder
  type: Peripheral |

<table>
<thead>
<tr>
<th>Search Term</th>
<th>Description</th>
<th>Syntax and Examples</th>
</tr>
</thead>
</table>
| types       | Return the item types supported by Search(). **Tip**: Call Describe() for any of the returned item types to get the fields and data types for that item type. | **Syntax**
  types: |

## Return Type

The Search() API returns the following objects:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource[]</td>
<td>The results of the search as a collection of Resource objects (see section Resource Hierarchy Description).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

## REST Protocol

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP Method</td>
<td><strong>GET</strong></td>
</tr>
<tr>
<td>Input/ Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><strong><a href="https://APPSRV01:8085/ResourceManagement/rest/resources?queryString=type%3aAgent+max%3a1+latest%3a1&amp;excludeFilter=">https://APPSRV01:8085/ResourceManagement/rest/resources?queryString=type%3aAgent+max%3a1+latest%3a1&amp;excludeFilter=</a></strong></td>
</tr>
</tbody>
</table>
### Sequence Diagram

**Figure 3-36 Sequence Diagram for Resource Management Search API**

#### Searching and Items with Pkey Maps

If your search returns a resource or membership that is mapped to more than one equipment instance, fields will include all fields for the earliest active equipment mapping. The pkey maps in `EquipmentMapping` contain the details for the other equipment instances.

This means that search results can be accessed successfully by legacy clients that do not expect pkey map details as well as by clients that are aware of the pkey map resource type.

Pkey maps are implemented internally as memberships, so if you search for all member types, the pkey map memberships will be visible in the search results along with the resource-to-resource memberships.

#### Describe

**Description**

The Describe API is used to return entity metadata for a passed in resource type. This metadata can then be used for constraining population of fields when performing edit or create operations on objects of that type.

**Parameters**

The `Describe()` API expects the following parameters:
<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>String[]</td>
<td>The resource type of the object for which the field metadata is required.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Return Type

The `Describe()` API returns the following objects:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceFieldMeta[]</td>
<td>A collection of <code>ResourceFieldMeta</code> objects containing metadata for the resource type passed in.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: For remote resource types and the member types, the returned metadata does not include fields that may be in a pkey map. To see these fields, you need to call <code>Describe()</code> and specify the corresponding resource or member pkey type.</td>
</tr>
<tr>
<td>ClusterResourceType</td>
<td>Future use. Defines the pkey elements that are relevant to the supplied cluster resource type.</td>
</tr>
</tbody>
</table>

### REST Protocol

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td><code>https://&lt;server&gt;:8085/ResourcesManagement/rest/resources/meta/&lt;resource type&gt;</code></td>
</tr>
<tr>
<td>HTTP Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><code>https://APPSRV01:8085/ResourcesManagement/rest/resources/meta/calling-search-space</code></td>
</tr>
</tbody>
</table>
Describing Items with Pkey Maps

If you describe a resource or membership that supports pkey maps, Describe() for the main item itself only returns the fields that are associated with the main item. To obtain the descriptions of the fields that may be returned on pkey maps, call Describe() for the corresponding resource or membership pkey type.

For example, to get the descriptions of all the fields for an Agent resource you will need to call Describe(Agent, AgentPkey).

Audit

Description

The Audit() API returns the audit records associated with one or more resource entities. This allows the client to display the activity associated with a resource, for example, identifying the reasons for an error status.

Parameters

The Audit() API expects the following parameters:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceKey[]</td>
<td>An array of resource identities for the items whose audits are to be returned.</td>
<td>Yes</td>
</tr>
<tr>
<td>Data Type</td>
<td>Description</td>
<td>Required?</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>String</td>
<td><strong>EventOutcome</strong>: A filter to reduce the returned audit records to those matching the outcome of the provisioning operation. One of: S: Success, F: Failure, N: Not applicable, U: Unknown or empty string to match all audit records.</td>
<td>Yes</td>
</tr>
<tr>
<td>DateTime</td>
<td><strong>fromDate</strong>: The start of the date range to filter the returned audit records. Note: for REST this is an 8601 encoded data string.</td>
<td>Yes</td>
</tr>
<tr>
<td>DateTime</td>
<td><strong>toDate</strong>: The end of the date range to filter the returned audit records. Note: for REST this is an 8601 encoded data string.</td>
<td>Yes</td>
</tr>
<tr>
<td>Int</td>
<td><strong>startIndex</strong>: Controls the pagination of audit records, specifies the index of the element at which to start.</td>
<td>Yes</td>
</tr>
<tr>
<td>Int</td>
<td><strong>resultsPerPage</strong>: Controls the pagination of audit records, specifies the number of elements to retrieve.</td>
<td>Yes</td>
</tr>
<tr>
<td>Bool</td>
<td><strong>summary</strong>: Whether the results are a summary or detailed results. If false, fills in the AdditionalData XML field in each of the returned audit records.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Return Type

The `Audit()` API returns the following object:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceAuditResults</td>
<td>The audit data for the resource id passed in.</td>
<td>ResourceAuditResults</td>
</tr>
</tbody>
</table>

### REST Protocol

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td><code>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/&lt;resource type&gt;/agents/315553/audits?fromDate=2011-10-04T00%3a00%3a00Z&amp;toDate=2012-06-06T10%3a43%3a57Z</code></td>
</tr>
<tr>
<td>HTTP Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input/ Output format</td>
<td>XML or JSON</td>
</tr>
</tbody>
</table>

### Example

- `https://APPSRV01:8085/ResourceManagement/rest/resources/agents/315553/audits?fromDate=2011-10-04T00%3a00%3a00Z&toDate=2012-06-06T10%3a43%3a57Z`
Sequence Diagram

**Figure 3-38  Sequence Diagram for Resource Management Audit API**

Auditing Resources with Pkey Maps

There are no special considerations when auditing resources or memberships that support pkey maps, since `ResourceAudit` objects do not contain pkey maps.

Upload

Description

The `Upload()` API is a generic API that allows binary data to be associated with an existing resource and uploaded to the appropriate remote system. If there is existing binary data associated with the specified resource, this data is replaced with the new binary data. The type of content being uploaded and any additional data required to specify the upload are determined by the resource type.


Parameters

The `Upload()` API expects the following parameters:

**Table 3-21  Parameters for Upload API**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
</table>

Cloud API Remote System Audit `auditResults` `audit`
Chapter 3: Resource Management Web Service

Web Service APIs

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceKey</td>
<td>The resource key of the resource to associate the data with.</td>
<td>Yes</td>
</tr>
<tr>
<td>String</td>
<td>The type of data being uploaded. Depends on the resource type (see Table 3-22).</td>
<td>Yes</td>
</tr>
<tr>
<td>AdditionalData</td>
<td>The additional data required for the upload. Depends on the resource type (see Table 3-22). This parameter is encoded in Part 1 of a MIME message.</td>
<td>Yes</td>
</tr>
<tr>
<td>Stream</td>
<td>The binary data to be uploaded and associated with the specified resource. This parameter is encoded in Part 2 of a MIME message.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

The supported resource types, and the associated content types and additional data required are:

**Table 3-22 Supported Resource Types for Upload API**

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Content Type</th>
<th>Content Format</th>
<th>Additional Data</th>
</tr>
</thead>
</table>
| Media file    | audio/wav    | .wav file to be associated with the specified media file resource | Servers
Comma separated list of URNs of the media file servers where the media file is to be uploaded to. If not present, the media file will be uploaded to all media file servers associated with the specified media file resource. No

**Return Type**

The Upload() API returns the following objects:

**Table 3-23 Return Type for Upload API**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestResult</td>
<td>The result of the upload.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
REST Protocol

**Table 3-24  REST Protocol URI for Upload API**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/&lt;resource type&gt;/&lt;id&gt;/content?contentType=&lt;contentType&gt;</td>
</tr>
<tr>
<td>HTTP Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/mediafiles/3565/content?contentType=audio/wav</td>
</tr>
</tbody>
</table>

See Table 3-25 for an example MIME message for this URL.

**Table 3-25  Example MIME Message for REST Protocol Upload API**

<table>
<thead>
<tr>
<th>Section</th>
<th>Example Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>Content-Type: multipart/form-data; boundary=&quot;-----------------------------8d08f053d6a90ad&quot;</td>
</tr>
<tr>
<td></td>
<td>Content-Length: 216829</td>
</tr>
<tr>
<td>Part 1: Additional Data</td>
<td>Content-Disposition: form-data; name=&quot;additionaldata&quot;</td>
</tr>
<tr>
<td>Format: XML or JSON</td>
<td>Content-Type: application/json; charset=utf-8</td>
</tr>
<tr>
<td></td>
<td>{&quot;Fields&quot;:[{&quot;Name&quot;:&quot;Servers&quot;,&quot;Value&quot;:&quot;21,25,29&quot;}]}</td>
</tr>
<tr>
<td>Part 2: Content</td>
<td>Content-Disposition: form-data; name=&quot;content&quot;; filename=&quot;datafile&quot;</td>
</tr>
<tr>
<td>Format: File</td>
<td>Content-Type: audio/wav</td>
</tr>
<tr>
<td></td>
<td>(file to upload follows as binary data, up to message length defined in header section).</td>
</tr>
</tbody>
</table>

See Table 3-25 for an example MIME message for this URL.
Download

Description

The `Download()` API is a generic API that returns the binary data that has been associated with a resource. The type of content being returned is determined by the resource type.

Parameters

The `Download()` API expects the following parameters:

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceKey</td>
<td>The identity of the resource to return the binary data for.</td>
<td>Yes</td>
</tr>
<tr>
<td>String</td>
<td>The type of binary data to be returned. Depends on the resource type (see Table 3-27).</td>
<td>Yes</td>
</tr>
</tbody>
</table>

The supported resource types, and the associated content types are:

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Content Type</th>
<th>Content Format</th>
</tr>
</thead>
</table>
### Resource Type

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Content Type</th>
<th>Content Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Media file</td>
<td>audio/wav</td>
<td>.wav file to be associated with the specified media file resource.</td>
</tr>
</tbody>
</table>

#### Return Type

The `Download()` API returns the following objects:

**Table 3-28  Return Type for Download API**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stream</td>
<td>The binary data that is associated with the resource. If the resource has no associated data, this value is null.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

#### REST Protocol

**Table 3-29  REST Protocol URI for Download API**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td><code>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/&lt;resource type&gt;s/&lt;id&gt;/content?contentType=&lt;contentType&gt;</code></td>
</tr>
<tr>
<td>HTTP Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><code>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/mediafiles/4893/content?contentType=audio/wav</code></td>
</tr>
</tbody>
</table>
Figure 3-40  Sequence Diagram for Resource Management Download API

Deploy

Description

The Deploy() API is a generic API that allows bulk update of one or more resources of a single type from a single binary deployment package. The deployment package contains both the information to identify each resource and the data to be associated with that resource. This data may include binary data to be uploaded to a remote system as well as other resource data.

If an associated resource already exists in CCMP, the existing data is replaced with the new data from the deployment package. If the associated resource does not yet exist, it is created.

The type of content being deployed and any additional data required to specify the deployment are determined by the resource type.


Parameters

The Deploy() API expects the following parameters:
Table 3-30 Parameters for Deploy API

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>The type of resource or resources being deployed.</td>
<td>Yes</td>
</tr>
<tr>
<td>String</td>
<td>The type of the deployment package. Depends on the resource type (see Table 3-31).</td>
<td>Yes</td>
</tr>
<tr>
<td>AdditionalData</td>
<td>Additional data required for the deployment. Depends on the resource type (see Table 3-31). This parameter is encoded in Part 1 of a MIME message.</td>
<td>Yes</td>
</tr>
<tr>
<td>Stream</td>
<td>The deployment package. This parameter is encoded in Part 2 of a MIME message.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

The supported resource types, and the associated content types and additional data required are:

Table 3-31 Supported Resource Types for Deploy API

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Content Type</th>
<th>Content Format</th>
<th>Additional Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>IVR Script</td>
<td>application/zip</td>
<td>ZIP file containing one or more VXML applications created by Cisco Call Studio</td>
<td>Folder ID where newly-created resources will be saved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Cluster ID of CVP Operations Console associated with the specified VXML application servers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Comma separated list of URNs of the VXML application servers where the package is to be deployed. If not present, the package will be deployed to all VXML application servers associated with the specified Cluster ID.</td>
</tr>
</tbody>
</table>

Return Type

The Deploy() API returns the following objects:
### Table 3-32  Return Type for Deploy API

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestResult[]</td>
<td>Array of results of the deployment.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### REST Protocol

#### Table 3-33  REST Protocol URI for Deploy API

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td><code>https://&lt;server&gt;:8085/ResourceManagement/resources/&lt;resource type&gt;/deploy?contentType=&lt;contentType&gt;</code></td>
</tr>
<tr>
<td>HTTP Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input/Output format</td>
<td>XML or JSON</td>
</tr>
<tr>
<td>Example</td>
<td><code>https://&lt;server&gt;:8085/ResourceManagement/rest/resources/ivr-scripts/deploy?contentType=application/zip</code></td>
</tr>
</tbody>
</table>

See Table 3-34 for an example MIME message for this URL.

#### Table 3-34  Example MIME Message for REST Protocol Deploy API

<table>
<thead>
<tr>
<th>Section</th>
<th>Example Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td><code>Content-Type: multipart/form-data; boundary=&quot;---------------------------8d0a38f0cd807bb&quot;</code></td>
</tr>
<tr>
<td></td>
<td><code>Content-Length: 24100</code></td>
</tr>
<tr>
<td>Part 1: Additional Data Format: XML or JSON</td>
<td><code>-----------------------------8d0a38f0cd807bb</code></td>
</tr>
<tr>
<td></td>
<td><code>Content-Disposition: form-data; name=&quot;additionaldata&quot;</code></td>
</tr>
<tr>
<td></td>
<td><code>Content-Type: application/json; charset=utf-8</code></td>
</tr>
<tr>
<td></td>
<td><code>{&quot;Fields&quot;: [{&quot;Name&quot;: &quot;FolderId&quot;, &quot;Value&quot;: &quot;c2a45666-6137-4109-9744-c82ca70429a&quot;}, {&quot;Name&quot;: &quot;Servers&quot;, &quot;Value&quot;: &quot;21,25,29&quot;}, {&quot;Name&quot;: &quot;ClusterId&quot;, &quot;Value&quot;: &quot;4356b97f-7466-44bd-ac6b-5de89594cd67&quot;}]</code></td>
</tr>
<tr>
<td>Part 2: Content Format: File</td>
<td><code>-----------------------------8d0a38f0cd807bb</code></td>
</tr>
<tr>
<td></td>
<td><code>Content-Disposition: form-data; name=&quot;content&quot;; filename=&quot;datafile&quot;</code></td>
</tr>
<tr>
<td></td>
<td><code>Content-Type: application/zip</code></td>
</tr>
<tr>
<td></td>
<td>(deployment package follows as binary data, up to message length defined in header section).</td>
</tr>
</tbody>
</table>
**Sequence Diagram**

![Sequence Diagram for Resource Management Deploy API](image)

**Figure 3-41** Sequence Diagram for Resource Management Deploy API

**Examples**

This section contains examples showing how to use the Resource Management Web Service APIs.

**Principles**

There several ways of using the APIs to achieve the same ends. For example, an agent may be added to an agent team or an agent team may have an agent added to it. Typically, for better performance, manipulate the parent object, which is always the first part of the membership name. So, in this example, the membership name is `Agent Agent Team Member` and the agent is the parent object.

The Search API is very powerful, but use the `max` and `offset` terms to page large data sets to avoid overloading the web service stack. The `childof` and `parentof` terms are also very useful to find current resource memberships. Use the `latest` term to find the current active memberships and for those systems using type-2 effective dating on resources such as agents.
Create Agent

Use Case Sequence

Prerequisites:
- The Create Tenant and Create Person use case sequences have been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral or peripherals that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the person record.</td>
</tr>
<tr>
<td>2</td>
<td>Select the folder in which the agent will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder.</td>
</tr>
<tr>
<td>3</td>
<td>Find all peripherals that support agents. By default these are peripherals with a client type of 30.</td>
</tr>
<tr>
<td>4</td>
<td>Create the agent linked to the person and peripheral and located in the selected folder. The request will be validated and then queued returning the new Agent URN.</td>
</tr>
<tr>
<td>5</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

SOAP Example

The following SOAP request creates a new agent called Agent1.

```xml
  <soap:Header/>
  <soap:Body>
    <ns1:Create>
      <ns1:resources>
        <!--Zero or more repetitions:-->  
        <ns1:Resource>
          <ns1:Type>Agent</ns1:Type>
          <ns1:EffectiveFrom>2009-01-01 00:00:00</ns1:EffectiveFrom>
          <ns1:EffectiveTo>2007-06-06 00:00:00</ns1:EffectiveTo>
          <ns1:Status>S</ns1:Status>
          <ns1:Fields>
            <!--Zero or more repetitions:-->  
          </ns1:Fields>
        </ns1:Resource>
      </ns1:resources>
    </ns1:Create>
  </soap:Body>
</soap:Envelope>
```
<res:NameValuePair>
    <res:Name>FolderId</res:Name>
    <res:Value>9F44B644-7C24-40CD-9DB9-6D1175D7FE3</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>PeripheralUrn</res:Name>
    <res:Value>-1</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>PersonUrn</res:Name>
    <res:Value>4568</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>Name</res:Name>
    <res:Value>Agent1</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>InternalName</res:Name>
    <res:Value>CICM.Agent1</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>Description</res:Name>
    <res:Value>Description of Agent1</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>FirstName</res:Name>
    <res:Value>Jim</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>LastName</res:Name>
    <res:Value>Smith</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>PeripheralNumber</res:Name>
    <res:Value>132456</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>PeripheralName</res:Name>
    <res:Value>Agent1</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>Supervisor</res:Name>
    <res:Value>True</res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>DomainName</res:Name>
    <res:Value>DOMAIN</res:Value>
</res:NameValuePair>
Update Agent

Use Case Sequence

Prerequisites:

- The Create Agent use case sequence has been executed without any existing agent desktop, skill groups and team memberships.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the agent desktop(s), skill group(s) and agent team(s) that the tenant administrator account allows.
- The Agent and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the agent record</td>
</tr>
<tr>
<td>2</td>
<td>Modify the field(s) of interest.</td>
</tr>
<tr>
<td>3</td>
<td>Update the agent. The agent resource status will go from Ready to Synchronizing until provisioned.</td>
</tr>
<tr>
<td>4</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

The agent re-skilling sequence where this agent is added as a member to a skill group is described in section Agent Re-Skilling.
SOAP Example

The following SOAP request will update the agent with identity 1234, setting their first name and last name to the passed in values.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Update>
      <res:resources>
        <res:Resource>
          <res:Identity>1234</res:Identity>
          <res:Type>Agent</res:Type>
          <res:EffectiveFrom>2009-01-01 00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06 00:00:00</res:EffectiveTo>
          <res:Status>S</res:Status>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>9F44B644-7C24-40CD-9DB9-6D1175DD7FE3</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>FirstName</res:Name>
              <res:Value>Jim</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>LastName</res:Name>
              <res:Value>Smith</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Update>
  </soap:Body>
</soap:Envelope>
```

Delete Agent

SOAP Example

An example of a SOAP request to the Delete API would be as follows: This request will delete the 2 agents with identities 17585 and 17586

```xml
  <soap:Header/>
  <soap:Body>
    <res:Delete>
      <res:resources>
        <res:Resource>
          <res:Identity>17585</res:Identity>
        </res:Resource>
        <res:Resource>
          <res:Identity>17586</res:Identity>
        </res:Resource>
      </res:resources>
    </res:Delete>
  </soap:Body>
</soap:Envelope>
```
Chapter 3: Resource Management Web Service

Examples

Retrieve Agent

SOAP Example

An example of a SOAP request to the Retrieve API to return two agent items from VW_DIM_AGENT with the related AgentUrns of 17585 and 17586.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Retrieve>
      <!--Optional:-->
      <res:resourceKeys>
        <!--Zero or more repetitions:-->-->
        <res:ResourceKey>
          <res:Identity>17585</res:Identity>
          <res:ResourceType>Agent</res:ResourceType>
        </res:ResourceKey>
        <res:ResourceKey>
          <res:Identity>17586</res:Identity>
          <res:ResourceType>Agent</res:ResourceType>
        </res:ResourceKey>
      </res:resourceKeys>
    </res:Retrieve>
  </soap:Body>
</soap:Envelope>
```

Search For Agent

SOAP Example

An example of a SOAP request calling the Search API to return the results all of the agents in the folder /Customer1 apart from the one with the ID 17585.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Retrieve>
      <!--Optional:-->
      <res:resourceKeys>
        <!--Zero or more repetitions:-->-->
        <res:ResourceKey>
          <res:Identity>17585</res:Identity>
          <res:ResourceType>Agent</res:ResourceType>
        </res:ResourceKey>
        <res:ResourceKey>
          <res:Identity>17586</res:Identity>
          <res:ResourceType>Agent</res:ResourceType>
        </res:ResourceKey>
      </res:resourceKeys>
    </res:Retrieve>
  </soap:Body>
</soap:Envelope>
```
Examples

Describing Agent and Person

SOAP Example

An example of a SOAP request calling the Describe API to return the metadata description of the agent and person.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Describe>
      <!-- Optional: -->
      <res:resourceType>
        <!-- Zero or more repetitions: -->
        <res:string>Agent</res:string>
      </res:resourceType>
      <!-- Zero or more repetitions: -->
      <res:resourceType>
        <!-- Zero or more repetitions: -->
        <res:string>Person</res:string>
      </res:resourceType>
    </res:Describe>
  </soap:Body>
</soap:Envelope>
```

Create Agent Team

Use Case Sequence

Prerequisites:
Chapter 3: Resource Management Web Service

- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral(s) that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the folder in which the agent team will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder. search type:folder latest:1 deleted:0</td>
</tr>
<tr>
<td>2</td>
<td>Find all peripherals that support agents and agent teams. By default these are peripherals with a client type of 30. search type:Peripheral property:ClientType=30 latest:1 deleted:0</td>
</tr>
<tr>
<td>3</td>
<td>Create the agent team linked to the peripheral and located in the selected folder. The request will be validated and then queued returning its new call type URN. create Agent Team</td>
</tr>
<tr>
<td>4</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve. search item &lt;Id&gt; type:&lt;ResourceType&gt; or retrieve Id = Agent Team</td>
</tr>
</tbody>
</table>

**SOAP Example**

The following SOAP request creates an agent team called “testTeam”.

```xml
<soap:Header/>  
<soap:Body>  
<res:Create>  
<!--Optional:-->  
<res:resources>  
<!--Zero or more repetitions:-->  
<res:Resource>  
<res:Identity>-1</res:Identity>  
<res:Type>Agent Team</res:Type>  
<res:EffectiveFrom>1900-01-01T00:00:00</res:EffectiveFrom>  
<res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>  
<res:Status>R</res:Status>  
<res:Changestamp>0</res:Changestamp>
</res:Resource>
</res:resources>
</res:Create>
</soap:Body>
</soap:Envelope>
```
Update Agent Team

Use Case Sequence

Prerequisites:

- The Create Agent Team use case sequence has been executed.
- The Agent Team and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the agent team record</td>
</tr>
<tr>
<td>2</td>
<td>Modify the field(s) of interest.</td>
</tr>
<tr>
<td>3</td>
<td>Update the agent team. The agent team resource status will go from Ready to Synchronizing until provisioned.</td>
</tr>
<tr>
<td>3</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>
Add/Remove Agent Team Members

Use Case Sequence

Prerequisites:
- The Create Agent Team use case sequence has been executed.
- The Agent Team and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the agent team record. This will have an associated peripheral parent.</td>
</tr>
<tr>
<td>2</td>
<td>Find the agents that are already associated with this Agent Team.</td>
</tr>
<tr>
<td>3</td>
<td>Find all the agents for this peripheral that are not already part of the agent team.</td>
</tr>
<tr>
<td>4</td>
<td>Remove unwanted agents from the agent team. The agent team resource status will go from Ready to Synchronizing until provisioned.</td>
</tr>
<tr>
<td>5</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
<tr>
<td>6</td>
<td>Add the required agents to the agent team. The agent team resource status will go from Ready to Synchronizing until provisioned.</td>
</tr>
<tr>
<td>7</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

Create Call Type

Use Case Sequence

Prerequisites:
- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral(s) that the tenant administrator account allows.
<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the folder in which the call type will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder.</td>
</tr>
<tr>
<td>2</td>
<td>Create the call type linked to the media routing domain and peripheral and located in the selected folder. The request will be validated and then queued returning its new call type URN.</td>
</tr>
<tr>
<td>3</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

**SOAP Example**

The following SOAP request creates a call type called “testCallType”.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!-- Optional: -->
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Update Call Type

Use Case Sequence

Prerequisites:

- The Create Call Type use case sequence has been executed without any existing agent memberships.
- The Call Type and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the Call Type record. retrieve Id = Call Type URN</td>
</tr>
<tr>
<td>2</td>
<td>Modify the field(s) of interest.</td>
</tr>
<tr>
<td>3</td>
<td>Update the call type. The call type resource status will go from Ready to Synchronizing until provisioned. update Call Type</td>
</tr>
</tbody>
</table>
### Add/Remove Routing Script Members

#### Use Case Sequence

**Prerequisites:**
- The Create Agent Team use case sequence has been executed.
- The Agent Team and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the Call Type record.</td>
</tr>
<tr>
<td>2</td>
<td>Find the routing scripts that are already associated with this Call Type.</td>
</tr>
<tr>
<td>3</td>
<td>Find all the routing scripts that are not already linked to this call type.</td>
</tr>
<tr>
<td>4</td>
<td>Remove unwanted routing scripts from the call type. The call type resource status will go from Ready to Synchronizing until provisioned.</td>
</tr>
<tr>
<td>5</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
<tr>
<td>6</td>
<td>Add the required routing scripts from step 3 to the agent team. The agent team resource status will go from Ready to Synchronizing until provisioned.</td>
</tr>
<tr>
<td>7</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

### Create Dialed Number

#### Use Case Sequence

**Prerequisites:**
The Create Tenant use case sequence has been executed.
The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the routing client(s) and media routing domain(s) that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the folder in which the dialed number will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder.</td>
</tr>
<tr>
<td>2</td>
<td>Find all relevant routing clients. By default these are peripherals with a client type of 13 (IVR) or 30 (PBX).</td>
</tr>
<tr>
<td>3</td>
<td>Find the relevant media routing domain; typically this is the default voice domain.</td>
</tr>
<tr>
<td>4</td>
<td>Create the dialed number linked to the media routing domain and routing client and located in the selected folder. The request will be validated and then queued returning its new Dialed Number URN.</td>
</tr>
<tr>
<td>5</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

**SOAP Example**

The following SOAP request creates a dialed number called “testDialedNumber”.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <res:Resource>
          <res:Identity>0</res:Identity>
          <res:Type>Dialed Number</res:Type>
          <res:EffectiveFrom>2010-03-11T12:19:37</res:EffectiveFrom>
          <res:EffectiveTo>2012-03-12T16:32:45.96</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
<res:Fields>
  <res:NameValuePair>
    <res:Name>FolderId</res:Name>
    <res:Value>240253c4-3f30-48c5-a3e6-65b3d94b100e</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>Name</res:Name>
    <res:Value>testDialedNumber</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>InternalName</res:Name>
    <res:Value>DN121</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>Description</res:Name>
    <res:Value></res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>RoutingClientUrn</res:Name>
    <res:Value>22049</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>MediaRoutingDomainUrn</res:Name>
    <res:Value>10793</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>Digits</res:Name>
    <res:Value>437457645</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>PermitApplicationRouting</res:Name>
    <res:Value>False</res:Value>
  </res:NameValuePair>
</res:Fields>

**Update Dialed Number**

**Use Case Sequence**

Prerequisites:
- The Create Dialed Number use case sequence has been executed without any existing agent memberships.
• The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the agents that the tenant administrator account allows.

• The Dialed Number and associated resources and members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the Dialed Number record. retrieve Id = Dialed Number URN</td>
</tr>
<tr>
<td>2</td>
<td>Modify the field(s) of interest.</td>
</tr>
<tr>
<td>3</td>
<td>Update the Dialed Number. The Dialed Number resource status will go from Ready to Synchronizing until provisioned. update Dialed number</td>
</tr>
<tr>
<td>4</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve. search item &lt;d&gt; type: &lt;ResourceType&gt; or retrieve Id = Dialed Number URN</td>
</tr>
</tbody>
</table>

Create Directory Number

Use Case Sequence

Prerequisites:

• The Create Tenant use case sequence has been executed.

• The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral or peripherals that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the folder in which the directory number will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder. search type: folder latest: 1 deleted: 0</td>
</tr>
<tr>
<td>2</td>
<td>Create the directory number located in the selected folder. The request will be validated and then queued returning its new directory number URN. create Directory Number</td>
</tr>
<tr>
<td>3</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve. search item &lt;d&gt; type: &lt;ResourceType&gt; or retrieve Id = Directory Number URN</td>
</tr>
</tbody>
</table>
SOAP Example

The following SOAP request creates a directory number called testDirectoryNumber.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:--> 
        <res:Resource>
          <!--Optional:--> 
          <res:Identity>-1</res:Identity>
          <res:Type>Directory Number</res:Type>
          <res:EffectiveFrom>1900-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <!--Zero or more repetitions:--> 
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>221C6722-B830-4848-9521-35B2DD8757D7</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Name</res:Name>
              <res:Value>testDirectoryNumber</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>DirectoryNumberOrPattern</res:Name>
              <res:Value>00786</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>IPCCEnabled</res:Name>
              <res:Value>true</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Update Directory Number

Use Case Sequence

Prerequisites:
- The Create Directory Number use case sequence has been executed without any existing IP Endpoint memberships.
- The Directory Number and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the Directory Number record retrieve Id = Call Type URN</td>
</tr>
<tr>
<td>2</td>
<td>Search for the IP Endpoint to which this Directory Number will be associated.. search type:&quot;IP Endpoint&quot; latest:1 status:R</td>
</tr>
<tr>
<td>3</td>
<td>Update the call type. The Directory Number resource status will go from Ready to Synchronizing until provisioned. create IP Endpoint Directory Number Member</td>
</tr>
<tr>
<td>4</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve. search item &lt;d&gt; type: &lt;ResourceType&gt; or retrieve Id = Directory Number URN</td>
</tr>
</tbody>
</table>

Create Folder

SOAP Example

The following SOAP request creates a folder called testfolder2 located under the folder with the ID 00000000-0000-0000-0000-000000000005 (Root).

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <res:Resource>
          <res:Identity>1</res:Identity>
          <res:Type>Folder</res:Type>
          <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Update Folder

SOAP Example

The update API may be used for the movement of items between different folders within the Unified CCMP database. Resource updates and move operations may not be performed within the same Update operation and should be achieved using 2 separate web service calls.

To move an item to a new folder the required fields of the resource should be passed in with new field called "NewFolderId". The value of the NewFolderId field should be set to the Unified CCMP identity for the folder to which the resource identity should be moved.

The following SOAP request moves a Dialed Number resource to another folder:

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
               xmlns="http://www.exony.com/schemas/2009/10/resourcemanagement">
  <soap:Header/>
  <soap:Body>
    <Update>
      <resources>
        <Resource>
          <Identity>23221</Identity>
          <Type>Dialed Number</Type>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>00000000-0000-0000-0000-000000000005</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Name</res:Name>
              <res:Value>test folder2</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Description</res:Name>
              <res:Value>test description</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>InheritPermissions</res:Name>
              <res:Value>true</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </Resource>
      </resources>
    </Update>
  </soap:Body>
</soap:Envelope>
```
The response from the request above is as follows:

```xml
  <Header/>
  <Body/>
</Envelope>
```

The response from the request above is as follows:

```xml
  <Header/>
  <Body/>
</Envelope>
```

The response from the request above is as follows:

```xml
  <Header/>
  <Body/>
</Envelope>
```

The response from the request above is as follows:

```xml
  <Header/>
  <Body/>
</Envelope>
```
The movement of this Dialed Number has caused a delete and create operation to be performed. This may happen if an item is moved to a folder outside of its current tenant or if an un-assigned item is moved to a new tenant. This behavior may be recognised by the identity contained within the response not matching that of the original request. The new identity should now be stored and used for all future web service changes relating to this item.

Create Group

Use Case Sequence

Prerequisites:

- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as a either the hoster or the tenant administrator.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Select the folder in which the group will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder.</td>
</tr>
<tr>
<td>2</td>
<td>Optionally find the security groups to which the new group will be added.</td>
</tr>
<tr>
<td>Action</td>
<td>API Calls and Parameters</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>3 Create the Group and optionally the Group Group members. The request will be executed synchronously returning its Group URN.</td>
<td><code>create Group Group Group Member x N</code></td>
</tr>
</tbody>
</table>

**SOAP Example**

The following SOAP request creates a group called `testGroup`.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>Group</res:Type>
          <res:EffectiveFrom>2001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>00000000-0000-0000-0000-000000000005</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Name</res:Name>
              <res:Value>testGroup</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Description</res:Name>
              <res:Value>test description</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Update Group

Use Case Sequence

Prerequisites:
- The Create Group use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the groups that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Retrieve the Group record.</td>
<td>retrieve Id = User ID</td>
</tr>
<tr>
<td>2 Modify the field(s) of interest.</td>
<td></td>
</tr>
<tr>
<td>3 Update the Group.</td>
<td>update Group</td>
</tr>
</tbody>
</table>

Create IP Endpoint

Use Case Sequence

Prerequisites:
- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral(s), Communication Managers that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Select the folder in which the IP Endpoint will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder.</td>
<td>search type: folder latest:1 deleted:0</td>
</tr>
<tr>
<td>2 Find all peripherals that support IP Endpoints. By default these are peripherals with a client type of 30.</td>
<td>search type: Peripheral property:ClientType=30 latest:1 deleted:0</td>
</tr>
<tr>
<td>3 Find the relevant Device Pool.</td>
<td>Search type: &quot;Device Pool&quot; latest:1 deleted:0</td>
</tr>
<tr>
<td>4 Find the relevant Calling Search Space. By default this is the one and only search space owned by this tenant on the single Communication Manager.</td>
<td></td>
</tr>
<tr>
<td>Action</td>
<td>API Calls and Parameters</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5  Find the Button template to be used by this IP Endpoint.</td>
<td></td>
</tr>
<tr>
<td>6  Find the Product Code for this IP Endpoint</td>
<td></td>
</tr>
<tr>
<td>7  Find the Protocol Code for this IP Endpoint.</td>
<td></td>
</tr>
<tr>
<td>8  Select the template data to be used for this phone.</td>
<td></td>
</tr>
<tr>
<td>9  Create the IP Endpoint linked to the Peripheral, Device pool, Calling Search Space, Button Template, Product Code, Protocol Code and located in the selected folder. The request will be validated and then queued returning its new IP Endpoint URN.</td>
<td></td>
</tr>
<tr>
<td>10 Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
<td>search item &lt;d&gt; type: &lt;ResourceType&gt; or retrieve Id = IP Endpoint URN</td>
</tr>
</tbody>
</table>

**SOAP Example**

The following SOAP request creates an IP Endpoint called testIPEndpoint.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:-->  
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>IP Endpoint</res:Type>
          <res:EffectiveFrom>1900-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:ChangeStamp>0</res:ChangeStamp>
          <res:Fields/>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
<res:NameValuePair>
    <res:Name>FolderId</res:Name>
    <res:Value>221C6722-B830-4848-9521-35B2DD8757D7</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>Name</res:Name>
    <res:Value>testIPEndpoint</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>PeripheralUrn</res:Name>
    <res:Value>3529</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>DevicePoolUrn</res:Name>
    <res:Value>17933</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>CallingSearchSpaceUrn</res:Name>
    <res:Value>17921</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>IPEndpointButtonTemplateUrn</res:Name>
    <res:Value>20553</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>ProtocolCodeBizUrn</res:Name>
    <res:Value>1769</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>ProductCodeBizUrn</res:Name>
    <res:Value>3321</res:Value>
</res:NameValuePair>

<res:NameValuePair>
    <res:Name>ExtensionMobilityEnabled</res:Name>
    <res:Value>false</res:Value>
</res:NameValuePair>

</res:Fields>
</res:Resource>
</res:resources>
</res:Create>
</soap:Body>
</soap:Envelope>
Update IP Endpoint

Use Case Sequence

Prerequisites:
- The Create IP Endpoint use case sequence has been executed without any existing agent memberships.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the agents that the tenant administrator account allows.
- The IP Endpoint and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Retrieve the IP Endpoint record.</td>
<td>retrieve Id = Skillgroup URN</td>
</tr>
<tr>
<td>2 Modify the field(s) of interest.</td>
<td></td>
</tr>
<tr>
<td>3 Update the IP Endpoint. The IP Endpoint resource status will go from Ready to Synchronizing until provisioned.</td>
<td>update IPEndpoint</td>
</tr>
<tr>
<td>4 Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
<td>search item &lt;d&gt; type: &lt;ResourceType&gt; or retrieve Id = IPEndpoint URN</td>
</tr>
</tbody>
</table>

Create Person

Use Case Sequence

Prerequisites:
- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as a either the hoster or the tenant administrator.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Select the folder in which the person will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder.</td>
<td>search type: folder latest: 1 deleted: 0</td>
</tr>
<tr>
<td>2 Create the Person in the required folder location. The request will be validated and then queued returning the new Person URN.</td>
<td>create Person</td>
</tr>
</tbody>
</table>
### SOA Example

The following SOAP request creates a Person record for a person called John Smith.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->  
      <res:resources>
        <!--Zero or more repetitions:-->  
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>Person</res:Type>
          <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <!--Zero or more repetitions:-->  
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>221C6722-B830-4848-9521-35B2DD8757D7</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Name</res:Name>
              <res:Value>JohnSmith</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>FirstName</res:Name>
              <res:Value>John</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>LastName</res:Name>
              <res:Value>Smith</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Update Person

Use Case Sequence

Prerequisites:

- The Create Person use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the groups that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Retrieve the Person record.</td>
<td>retrieve Id = Person ID</td>
</tr>
<tr>
<td>2 Modify the field(s) of interest.</td>
<td></td>
</tr>
<tr>
<td>3 Update the Person.</td>
<td>update Person</td>
</tr>
</tbody>
</table>

Create Skill Group

Use Case Sequence

Prerequisites:

- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral(s) that the tenant administrator account allows.
### Action | API Calls and Parameters
--- | ---
1. Select the folder in which the skill group will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder. | search type: folder latest: 1 deleted: 0
2. Find all peripherals that support skill groups. By default these are peripherals with a client type of 30. | search type: Peripheral property: ClientType=30 latest: 1 deleted: 0
3. Find the relevant media routing domain; typically this is the default voice domain | Search type: "Media Routing Domain" cstext: "Cisco_Voice" latest: 1 deleted: 0
4. Create the skill group linked to the media routing domain and peripheral and located in the selected folder. The request will be validated and then queued returning its new Skill Group URN. | create Skillgroup
5. Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve. | search item <Id> type: <ResourceType> or retrieve Id = Skillgroup URN

### SOAP Example
The following SOAP request creates a skill group called testSkill.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:-->
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>SkillGroup</res:Type>
          <res:EffectiveFrom>1900-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <res:Field/>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Update Skill Group

Use Case Sequence

Prerequisites:
- The Create Skill Group use case sequence has been executed without any existing agent memberships.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the agents that the tenant administrator account allows.
The Skill group and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Retrieve the skill group record</td>
<td>retrieve Id = Skill group URN</td>
</tr>
<tr>
<td>2 Modify the field(s) of interest.</td>
<td>update Skill Group</td>
</tr>
<tr>
<td>3 Update the skill group. The skill group resource status will go from Ready to Synchronizing until provisioned.</td>
<td>update Skill Group</td>
</tr>
<tr>
<td>4 Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
<td>search item &lt;d&gt; type: &lt;ResourceType&gt; or retrieve Id = Skill group URN</td>
</tr>
</tbody>
</table>

The agent re-skilling sequence where this agent is added as a member to a skill group is described in section Agent to Skill Group Membership.

Create Tenant

Use Case Sequence

Prerequisites:
- The caller of the Unified CCMP Web Services is logged in as the host administrator.
- The underlying equipment is located in the folder structure.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Select the equipment clusters that this tenant will be placed in.</td>
<td>search type: &quot;Cluster Resource&quot; property: &quot;ResourceType internal Name&quot; =CRT_CICM and/or search type: &quot;Cluster Resource&quot; property: &quot;ResourceType internal Name&quot; =CRT_CCM</td>
</tr>
<tr>
<td>2 Create the tenant, setting the MappedClusterResources field to the comma separated cluster resources found in step 1. The request will be validated and then queued, returning its new Tenant URN.</td>
<td>create Tenant</td>
</tr>
<tr>
<td>3 Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
<td>search item &lt;TenantUrn&gt; or retrieve Id=TenantUrn</td>
</tr>
</tbody>
</table>
### Update Tenant

**Use Case Sequence**

**Prerequisites:**
- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as either the host administrator or the tenant administrator.
- The Tenant and associated resources/members are all in the Ready state.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the Tenant record. retrieve Id = Tenant URN</td>
</tr>
<tr>
<td>2</td>
<td>Modify the field(s) of interest.</td>
</tr>
<tr>
<td>3</td>
<td>Update the Tenant. The Tenant resource status will go from Ready to Synchronizing until provisioned. update Tenant</td>
</tr>
<tr>
<td>4</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve. search item &lt;d&gt; type: &lt;ResourceType&gt; or retrieve Id = Tenant URN</td>
</tr>
</tbody>
</table>

### Create User

**Use Case Sequence**

**Prerequisites:**
- The Create Tenant use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as either the hoster or the tenant administrator.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Create one or more tenant users under the new tenant (see Create User use case sequence in Create User). This would typically be one or more Tenant administrator accounts. Tip. The FolderId field for the User can be determined by retrieving the Tenant record in step 3 or by supplying the path text which will be “/” + tenant name set in step 2. create User</td>
</tr>
</tbody>
</table>
### Chapter 3: Resource Management Web Service

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the folder in which the user will be placed. This is typically either the tenant folder or a sub folder underneath the tenant folder.</td>
<td>search type: folder latest: 1 deleted: 0</td>
</tr>
<tr>
<td>2. Optionally find the security groups to which the new user will be added.</td>
<td>search type: Group enabled: 1 folder: /&lt;tenant&gt;**</td>
</tr>
<tr>
<td>3. Create the User and optionally the User Group members. The request will be executed synchronously returning its User URN.</td>
<td>create User User Group Member xN</td>
</tr>
</tbody>
</table>

### SOAP Example

The following SOAP request creates a user called testUser.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>User</res:Type>
          <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>00000000-0000-0000-0000-000000000004</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Description</res:Name>
              <res:Value>testUser description</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>LoginName</res:Name>
              <res:Value>testUser</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Time Zones

Users can be created using the default server time-zone information (as in the request above), or they can optionally be created with the time-zone specified at creation time. To specify the time zone in the create request, add the optional parameter TimeZone and supply the name of the time zone.

The list of supported time zones is contained in the database table TE_ADM_TIME_ZONE and is reproduced in Appendix C.

Update User

Use Case Sequence

Prerequisites:
- The Create User use case sequence has been executed.
- The caller of the Unified CCMP Web Services is logged in as the tenant administrator and only has access to the groups that the tenant administrator account allows.

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Retrieve the User record.</td>
<td>retrieve Id = User ID</td>
</tr>
<tr>
<td>2 Modify the field(s) of interest.</td>
<td></td>
</tr>
<tr>
<td>3 Update the User</td>
<td>update User</td>
</tr>
</tbody>
</table>

**Agent to Skill Group Membership**

**SOAP Example**

The following SOAP request creates a membership between an agent and a skillgroup where the identity of the agent is 7565 and the identity of the skillgroup is 17701.
Agent to Agent Team Membership

**SOAP Example**

The following SOAP request creates a membership between an agent and an agent team where the identity of the agent is 7565 and the identity of the agent team is 7917.
  <soap:Header/>
  <soap:Body>
    <res:create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:-->  
      <res:Resource>
        <res:Identity>-1</res:Identity>
        <res:Type>Agent Agent Team Member</res:Type>
        <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
        <res:EffectiveTo>2079-06-06T00:00:00</res:EffectiveTo>
        <res:Status>R</res:Status>
        <res:Changestamp>0</res:Changestamp>
        <res:Fields>
          <!--Zero or more repetitions:-->
          <res:NameValuePair>
            <res:Name>ParentId</res:Name>
            <res:Value>7917</res:Value>
          </res:NameValuePair>
          <res:NameValuePair>
            <res:Name>ChildId</res:Name>
            <res:Value>7565</res:Value>
          </res:NameValuePair>
          <res:NameValuePair>
            <res:Name>Supervisor</res:Name>
            <res:Value>false</res:Value>
          </res:NameValuePair>
          <res:NameValuePair>
            <res:Name>PrimarySupervisor</res:Name>
            <res:Value>false</res:Value>
          </res:NameValuePair>
          <res:NameValuePair>
            <res:Name>PhysicalMember</res:Name>
            <res:Value>true</res:Value>
          </res:NameValuePair>
        </res:Fields>
      </res:Resource>
    </res:resources>
    </res:create>
  </soap:Body>
</soap:Envelope>
IP Endpoint to Directory Number Membership

SOAP Example

The following SOAP request creates a membership between an IP endpoint and a directory number where the identity of the IP endpoint is 20765 and the identity of the directory number is 18697.
    <soap:Header/>
    <soap:Body>
        <res:Create>
            <!--Optional:-->
            <res:resources>
                <!--Zero or more repetitions:-->  
                <res:Resource>
                    <res:Identity>-1</res:Identity>
                    <res:Type>IP Endpoint Directory Number Member</res:Type>
                    <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
                    <res:EffectiveTo>2079-06-06T00:00:00</res:EffectiveTo>
                    <res:Status>R</res:Status>
                    <res:Changestamp>0</res:Changestamp>
                    <res:Fields>
                        <!--Zero or more repetitions:-->  
                        <res:NameValuePair>
                            <res:Name>ParentId</res:Name>
                            <res:Value>18697</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>ChildId</res:Name>
                            <res:Value>20765</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>LineIndex</res:Name>
                            <res:Value>0</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>MaxNumCalls</res:Name>
                            <res:Value>2</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>BusyTrigger</res:Name>
                            <res:Value>1</res:Value>
                        </res:NameValuePair>
                    </res:Fields>
                </res:Resource>
            </res:resources>
        </res:Create>
    </soap:Body>
</soap:Envelope>
User to Group Membership

SOAP Example

The following SOAP request adds the user with the identity 80479952-4236-456d-9dc4-668a14a9f8a9 to the group with the identity 7d80aa2f-d649-40d8-9c79-0036352d64d2.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:--> 
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>User Group Member</res:Type>
          <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <!--Zero or more repetitions:--> 
            <res:NameValuePair>
              <res:Name>ParentId</res:Name>
              <res:Value>7d80aa2f-d649-40d8-9c79-0036352d64d2</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>ChildId</res:Name>
              <res:Value>80479952-4236-456d-9dc4-668a14a9f8a9</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
        </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Group to Group Membership

SOAP Example

The following SOAP request adds the group with the identity 7d80aa2f-d649-40d8-9c79-0036352d64d2 to the group with the identity bf272315-0435-49bd-a117-d2117754593a.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:-->  
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>Group Group Member</res:Type>
          <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00.0000000</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <!--Zero or more repetitions:-->  
            <res:NameValuePair>
              <res:Name>ParentId</res:Name>
              <res:Value>bf272315-0435-49bd-a117-d2117754593a</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>ChildId</res:Name>
              <res:Value>7d80aa2f-d649-40d8-9c79-0036352d64d2</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
Dialed Number to Call Type Membership

SOAP Example

The following SOAP request associates the Dialed Number with the identity 9149 to the Call Type with the identity 8909. Note that the Dialed Number Call Type Membership has a number of properties that may be passed in during create and edit operations.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <res:Resource>
          <res:Identity>0</res:Identity>
          <res:Type>Dialed Number Call Type Member</res:Type>
          <res:EffectiveFrom>2010-03-11T12:19:37</res:EffectiveFrom>
          <res:EffectiveTo>2012-03-12T16:32:45.96</res:EffectiveTo>
          <res:Status>S</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>ParentId</res:Name>
              <res:Value>8909</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>ChildId</res:Name>
              <res:Value>9149</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Item</res:Name>
              <res:Value>0</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>ClinetWildCardType</res:Name>
              <res:Value>5</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
The Dialed Number Call Type membership has a number of fields available on it and may be configured in different ways for Dialed Numbers that belong to Voice or non-Voice Media Routing Domains.

The fields that may be set when creating or updating a Dialed Number Call Type Membership are:

- **ParentId** (required) the identity of the Call Type resource
- **ChildId** (required) the identity of the Dialed Number resource
- **RegionUrn**
- **Item** (required)
- **CLIWildCard**
- **CLIWildCardType** (required)
- **CEDWILDCard**
- **CEDWILDCardType** (required)
- **Description**

When creating or editing a Dialed Number connected to a Voice Media Routing Domain, the configuration may be specified as follows.

**Call Line ID Configuration**

<table>
<thead>
<tr>
<th>Call Line ID</th>
<th>CLIWildCardType</th>
<th>CLIWildCard</th>
<th>RegionUrn</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>5</td>
<td>Null</td>
<td>Null</td>
</tr>
<tr>
<td>Region</td>
<td>CSG.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prefix</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Match</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Call Line ID</th>
<th>CLIWildCardType</th>
<th>CLIWildCard</th>
<th>RegionUrn</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>4</td>
<td>Null</td>
<td>The Identity of the selected Region</td>
</tr>
<tr>
<td>Region</td>
<td>CSG.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prefix</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Match</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Call Line ID</th>
<th>CLIWildCardType</th>
<th>CLIWildCard</th>
<th>RegionUrn</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>6</td>
<td>Prefix1</td>
<td>Null</td>
</tr>
<tr>
<td>Region</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prefix</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Match</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Call Entered Digits Configuration

When creating or editing a Dialed Number connected to a non-Voice Media Routing Domain the configuration may be specified as follows.
### Agent Re-Skilling

The following example describes the Resource Management Web Service API usage when performing a re-skilling action from a third party client.
The example is a third party application that first lists Skill Groups that the user Bob may see. Upon selection of one of those Skill Groups a list of Agents that are currently working in that Skill is presented along with a list of Agent available to work in that Skill that Bob can see. Agents may be removed and added to and from the Skill Group and when complete the "Save" button may be pressed to save the changes to the Unified CCMP Database and the remote system.

The web service API interaction for this example is as follows:

**Step 1**
Retrieve the list of Skill Groups that the user can see.

Call to Search API passing in the search string:

```
type:SkillGroup folder:/Tenant latest:1 max:50 status:R
```

**Step 2**
The user selects the Skill Group that they wish to change (for example, identity 1234) and is presented with the list of Agents already in that Skill Group

Call to the Search API with the search string

```
childof:SkillGroup,1234>[Status=R;Deleted=0] type:Agent latest:1 max:50
```

**Step 3**
The Peripheral identity of the Peripheral that the Skill Group is on is retrieved

Call to the Search API with the search string:

```
parentof:SkillGroup,1234 type:Peripheral
```

**Step 4**
The Peripheral identity (4321) is used to get the list of possible Agents that may be added to the Skill Group

Call to Search API with the search string

```
childof:Peripheral,4321 type:Agent
```

The API is also passed a collection of ResourceKeys of the selected Agents to exclude them from the list

**Step 5**
The user uses the client to create and remove Skill Group/Agent members as they require. The user presses the "Save" button to commit the change to Unified CCMP Database.

**Step 6**
The 'to delete existing memberships' and 'to add new memberships' collections built up with the previous search queries are maintained by the client. A call to the Delete API is then made passing the 'to delete collection' of Resource objects with the same
type set and the parent/child identities of the relationships to be removed. A subsequent call to the Create API is made, passing the 'to add' collection of Resource objects with the type AgentSkillGroupMember and the parent and child identities set accordingly.

Unified CCMP performs a number of validation and capacity checks when performing provisioning requests. It is recommended when provisioning resource memberships that Delete procedures are performed before create to reduce the risk of capacity exceptions occurring.

User Creation

The following example describes the usage of the Web Service APIs to create a new Unified CCMP user account. This account will be able to perform item create/edit and delete operations for a specific tenant.

When a new tenant is created in Unified CCMP (for example, when a new Unified CCE Customer Definition is imported from Unified CCE), 3 security groups are automatically configured to allow for simplistic configuration of security for newly added users.

The automatically created groups are:
- Advanced Users (high level access, intended for tenant administrator accounts)
- Basic Users (low level access, intended for reporting only accounts)
- Supervisor Users (medium level access, intended for supervisors who require access to reports and permissions to re-skill agents).

An example showing the web service API interactions to create a tenant admin account is as follows:

**Step 1**

Retrieve the Folder ID for the for the tenant folder. Call the Search API with the following search string:

type:IT_FOLDER folder:/ name:"v723"

where v723 is the name of the tenant. The returning items identity field will be the FolderID that will be used later

**Step 2**

Create a new user in the tenant folder. Call the Create API specifying User as the Type, F o l d e r I D as the folder id retrieved above and Name/Logi nName as the name of the user that is to be created. An initial password must also be specified. The create API will return the identity of the new user if successful which will be used later

**Step 3**

Locate the Advanced Users group for the tenant that we have created the new user in. Call the Search API with the following search string:
Chapter 3: Resource Management Web Service

Examples

Type:IT_GROUP folder:”221c6722-b830-4848-9521-35b2dd8757d7" name:"Advanced Users"

where 221c6722-b830-4848-9521-35b2dd8757d7 is the identity for the tenant's folder retrieved in step 1.

**Step 4**
Create a new User/Group member between the new user and the Advanced Users group for the users tenant. Call the Create API specifying UserGroupMember as the Type, the new user id retrieved in step 2 as the ChildId and the group id retrieved in step 3 as the ParentId.

**Step 5**
The new user has been created and added to the Advanced Users group.

**Move Unallocated Dialed Number to Tenant Folder**
An example showing the web service API interactions to move an unallocated dialed number(s) to a pre-created call type/routing script is as follows:

**Step 1**
Get the destination Tenant folder id. Call the Search API with the following search string:

type:IT_FOLDER folder:/ name:"v723"

where v723 is the name of the tenant.

The returning items identity field will be the FolderID that will be used later.

**Step 2**
Get the list of unallocated Dialed Numbers that the tenant login can see. Note: these are typically located either in the /Unallocated/<equipment> folder (by default on a multi-tenant Unified CCE) or manually moved to a product offering specific folder in /Shared, for example, /Shared/SFDC/Bronze.

Call to Search API passing in the search string

type:"Dialed Number" folder:"<unallocated DNs folder>" max:"<desired number> offset:0 latest:1 deleted:0

For example,

type:"Dialed Number" folder:"/Shared/SFDC/Bronze" max:3 offset:0 latest:1 deleted:0

**Step 3**
Move the returned dialed numbers to the required tenant folder. For each dialed number in the returned array set its folder id to the tenant folder id found in step 1. Call Update AI with the modified dialed numbers.
Step 4
The Dialed Numbers have been moved from the unallocated location to the tenant folder.

Link Dialed Number to Call Type/Routing Script
Mapping the call treatment for a Dialed Number is ensuring that the Dialed Number is linked to the correct Call Type which in turn is linked to the correct Routing Script which contains the call treatment logic. This use case assumes that the Dialed Number has been moved to the correct tenant folder and that the Call Type(s) are already associated with Routing Script(s).

The web service API interactions to link an dialed number in a tenant folder to an existing call type/routing script would be as follows:

Step 1
Get the Dialed Numbers anywhere in the tenant specific folder structure. Call to Search API passing in the search string:

type:"Dialed Number" folder:/<tenant location>**" max:<desired number> offset:0 latest:1 deleted:0 status:R

For example,

type:"Dialed Number" folder:/Acme**" max:5 offset:0 latest:1 deleted:0

Step 2
Get the Call Types anywhere in the tenant specific folder structure.
Call to Search API passing in the search string:

type:"Call Type" folder:/<tenant location>**" max:<desired number> offset:0 latest:1 deleted:0

For example,

type:"Call Type" folder:/Acme**" max:5 offset:0 latest:1 deleted:0

Step 3
Add the relevant dialed number(s) to the relevant call type(s) as shown in section Dialed Number to Call Type Membership.

Call the Create API for resource type "Dialed Number Call Type Member" with parent and child item urns set to the relevant Call Type and Dialed Number urns found in the previous two searches.

Step 4
The new dialed number call type memberships will be added to the Unified CCMP Database and their membership URNs returned to the caller for tracking purposes.
Step 5

To check for any existing memberships between call types and dialed numbers then the following Search APIs can be called.

To find if a call type has any associated dialed numbers either

childof: "Call Type", <call type urn> type: "Dialed Number" latest: 1

or

memberbyparent: "Call Type", <call type urn>, "Dialed Number Call Type Member"

To find if a dialed number has any associated call types, either

parentof: "Dialed Number", <dialed number urn> type: "Call Type" latest: 1

or

memberbychild: "Dialed Number", <dialed number urn>, "Dialed Number Call Type Member"

Unlink Dialed Number to Call Type/Routing Script

This example details the steps required to remove a membership between a Dialed Number and Call Type; for example when the Dialed Number is to be moved to a different Call Type. The example assumes that the Dialed Number is already linked to the Call Type and that both the resources and their membership is in the Ready state (that is, they can be provisioned). The web service API interactions in this case would be as follows:

Step 1

Get the Dialed Numbers anywhere in the tenant specific folder structure.

Call to Search API passing in the search string:

type: "Dialed Number" folder: /<tenant location>**" cstext: <dialed number name> status: R latest: 1

for example,

type: "Dialed Number" folder: /Acme**" cstext: "CCMIST_RC1.2551" status: R latest: 1

Step 2

Find its existing Dialed Number Call Type Membership(s).

Call to Search API passing in the search string:

memberbychild: "Dialed Number", <dialed number urn>, "Dialed Number Call Type Member" latest: 1 status: R
Step 3
Delete the membership(s).
Call the Delete API for resource type "Dialed Number Call Type Member" with resource type "Dialed Number Call Type Member" and Identity set to the identities found in the previous membership search.

Step 4
The existing dialed number call type memberships will be marked for deletion in the Unified CCMP Database and will be provisioned in due course. Note that the memberships will be purged from the underlying Unified CCE in this case but will remain in the Unified CCMP Database with a status of Deleted for reporting and tracking purposes.

Step 5
Subscribers who have subscribed to this resource type will receive a notification for each step in the state machine cycle until the operation is either successful or fails provisioning.

Rename Resources
This example details the steps required to rename the resources for a specific tenant in the underlying equipment. This would typically be done when activating a template or 'blank' customer in which all the resources are pre-allocated with a template prefix name and is done for operational tracking and having meaningful names in the end user reports. The example assumes that all the resources and their membership are in the Ready state, that is, they can be provisioned. The web service API interactions in this case would be as follows:

Step 1
Get the resources underneath the tenant:
Call to Search API passing in the search string:
folder:/<tenant folder>** status:R latest:1 max:50 offset:0
This will return the all the resources under the tenant folders. Those that are useful to rename for reporting are Tenant, Call Types and Skill Groups. Routing Scripts, Network Vru Scripts and Routes may be usefully renamed for operation tracking. Several calls may be needed using the max and offset keywords to page the data.

Step 2
Rename each resource returned in the previous (including the tenant resource):
Call the Update API changing both the resources internal name and name fields to the desired tenant template name. Typically each resource type has its own naming convention that is specific to each installation
For example; a pattern for Network VRU Scripts may be <Tenant Name>_N_VRU and the caller could simply rename the tenant prefix, for example,
Blank103.1.1.VRU to Acme.1.1.VRU. The rename mask should ensure that it matches the rules for resource names in terms of length and legal characters.

Create Agent with Pkey Maps, Method 1 (Explicit Pkey Maps)

Description

This example creates an Agent that is linked to two different remote equipment instances. In this example, the pkey map is specified explicitly and separately for each remote equipment instance.

Use separate pkey maps if an item has different details on different remote equipment instances. You can mix this method with method 2 (implicit pkey maps).

You can specify some fields with the main item as common fields for a list of equipment instances and specify some fields individually for each equipment instance as pkey maps.

Use Case Sequence

Prerequisites:

- The Create Tenant and Create Person use case sequences have been executed.
- The caller of the CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral or peripherals that the tenant administrator account allows.

Table 3-35 Create Agent with Pkey Maps, Method 1 (Explicit Pkey Maps)

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create the Agent, specifying the Agent details and pkey maps as in the SOAP example below.</td>
</tr>
<tr>
<td>2</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

SOAP Example

The following SOAP request creates an agent, with first name “Jim” and last name “Smith”, that is linked to two agents (“Jim Smith A” and “Jim Smith B”) on two remote equipment instances.

In this example, two pkey maps are specified, one for each equipment instance.
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <res:Resource>
          <res:Type>Agent</res:Type>
          <res:EffectiveFrom>2013-01-01 00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06 00:00:00</res:EffectiveTo>
          <res:Status>S</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>9F44B644-7C24-40CD-9DB9-6D1175D7FE3</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>PeripheralUrn</res:Name>
              <res:Value>-1</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>PersonUrn</res:Name>
              <res:Value>4568</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>FirstName</res:Name>
              <res:Value>Jim</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>LastName</res:Name>
              <res:Value>Smith</res:Value>
            </res:NameValuePair>
          </res:Fields>
          <res:EquipmentMapping>
            <res:Equipment>
              <res:Type>Agent Pkey</res:Type>
              <res:Fields>
                <res:NameValuePair>
                  <res:Name>Name</res:Name>
                  <res:Value>Jim Smith A</res:Value>
                </res:NameValuePair>
                <res:NameValuePair>
                  <res:Name>InternalName</res:Name>
                  <res:Value>/</res:Value>
                </res:NameValuePair>
              </res:Fields>
            </res:Equipment>
          </res:EquipmentMapping>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
<res:NameValuePair>
    <res:Name>Description</Name>
    <res:Value>Jim Smith on Equipment</Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>ClusterResourceId</Name>
    <res:Value>{c5ed5103-55cc-420b-8fa9-9d64c2ea339}</Value>
</res:NameValuePair>
</res:Fields>
</res:Equipment>
</res:EquipmentMapping>

9d64c2ea339<Val ue>

<res:NameValuePair>
    <res:Name>Agent_Pkey</Name>
    <res:Value>Jim Smith B</Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>InternalName</Name>
    <res:Value/></res:Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>Description</Name>
    <res:Value>Jim Smith on Equipment B</Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>DomainLoginName</Name>
    <res:Value>Jim</Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>ClusterResourceId</Name>
    <res:Value>{5103c5ed-d43e-fa4e-98f0-2eaa33a9d64c}</Value>
</res:NameValuePair>
</res:Fields>
</res:Equipment>
</res:EquipmentMapping>

B<Val ue>

<res:NameValuePair>
    <res:Name>DomainLoginName</Name>
    <res:Value>Jim</Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>ClusterResourceId</Name>
    <res:Value>{5103c5ed-d43e-fa4e-98f0-2eaa33a9d64c}</Value>
</res:NameValuePair>
</res:Fields>
</res:Equipment>
</res:EquipmentMapping>

2eaa33a9d64c<Val ue>

<res:NameValuePair>
    <res:Name>DomainLoginName</Name>
    <res:Value>Jim</Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>ClusterResourceId</Name>
    <res:Value>{5103c5ed-d43e-fa4e-98f0-2eaa33a9d64c}</Value>
</res:NameValuePair>
</res:Fields>
</res:Equipment>
</res:EquipmentMapping>

A<Val ue>

<res:NameValuePair>
    <res:Name>Description</Name>
    <res:Value>Jim Smith on Equipment</Value>
</res:NameValuePair>
<res:NameValuePair>
    <res:Name>ClusterResourceId</Name>
    <res:Value>{c5ed5103-55cc-420b-8fa9-9d64c2ea339}</Value>
</res:NameValuePair>
</res:Fields>
</res:Equipment>
</res:EquipmentMapping>

9d64c2ea339<Val ue>
Create Agent Team with Pkey Maps, Method 2 (Implicit Pkey Maps)

Description

This example creates an Agent Team that is linked to two different remote equipment instances. In this example, the remote equipment instances are specified in a comma-separated list as part of the main Agent Team item. When the Agent Team is created, a pkey map containing the relevant data will be created for each remote equipment instance.

Tip

Use this method if an item has the same details on different remote equipment instances. You can mix this method with method 1 (explicit pkey maps). You can specify some fields with the main item as common fields for a list of equipment instances and specify some fields individually for each equipment instance as pkey maps.

Use Case Sequence

Prerequisites:

- The Create Tenant use case sequence has been executed.
- The caller of the CCMP Web Services is logged in as the tenant administrator and only has access to the peripheral or peripherals that the tenant administrator account allows.

Table 3-36 Create Agent Team, Two Pkey Maps, Method 2 (Combined Pkey Maps)

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create the Agent Team, specifying the Agent Team details and pkey maps as in the SOAP example below.</td>
</tr>
<tr>
<td>2</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

SOAP Example

The following SOAP request creates an Agent Team, “Test Agent Team”, that is linked to two Agent Teams on two remote equipment instances, where the two Agent Teams have identical details on both equipment instances.

In this example, the remote equipment instances are specified in a comma-separated list.

```xml
```
Update Agent (Additional Pkey Map)

Description

This example adds an additional remote equipment mapping to an existing Agent.

Use Case Sequence

Prerequisites:
- The specified Agent already exists.
Table 3-37  Update Agent (Additional Remote Equipment Mapping)  
Use Case

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the agent record</td>
</tr>
<tr>
<td></td>
<td>retrieve Id = Agent Id</td>
</tr>
<tr>
<td>2</td>
<td>Modify the agent record to include a pkey map containing the mapping to the additional remote equipment.</td>
</tr>
<tr>
<td>3</td>
<td>Update the agent, specifying the remote equipment mapping in a pkey as in the SOAP example below.</td>
</tr>
<tr>
<td>4</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
<tr>
<td></td>
<td>search item &lt;Id&gt;</td>
</tr>
<tr>
<td></td>
<td>type: &lt;ResourceType&gt;</td>
</tr>
<tr>
<td></td>
<td>or</td>
</tr>
<tr>
<td></td>
<td>retrieve Id = Agent Id</td>
</tr>
</tbody>
</table>

SOAP Example

The following SOAP request updates the agent with identity 1234 to include a link to an additional remote equipment instance.

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  <soap:Header/>
  <soap:Body>
    <res:Update>
      <res:resources>
        <res:Resource>
          <res:Identity>1234</res:Identity>
          <res:Type>Agent</res:Type>
          <res:Fields>
          </res:Fields>
          <res:EquipmentMapping>
            <res:Equipment>
              <res:Type>Agent Pkey</res:Type>
              <res:Fields>
                <res:NameValuePair>
                  <res:Name>Name</res:Name>
                  <res:Value>Jim Smith X</res:Value>
                </res:NameValuePair>
                <res:NameValuePair>
                  <res:Name>Internal Name</res:Name>
                  <res:Value>/</res:Value>
                </res:NameValuePair>
              </res:Fields>
            </res:Equipment>
          </res:EquipmentMapping>
        </res:Resource>
      </res:resources>
    </res:Update>
  </soap:Body>
</soap:Envelope>
```
Update Agent (Delete a Pkey Map)

Description

This example deletes one of the remote equipment mappings for an Agent.

You cannot delete the last remaining remote equipment mapping unless you delete the Agent.

Use Case Sequence

Prerequisites:
- The specified Agent already exists and has at least two pkey map items.

Table 3-38 Update Agent (Additional Remote Equipment Mapping)

<table>
<thead>
<tr>
<th>Use Case</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Action</strong></td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>Action</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| If required, wait for the asynchronous                                | search item <d>
| notification if a subscription has been set up (SOAP only), or begin  | type: <ResourceType>
| a poll sequence using search or retrieve.                              | or retrieve Id = Agent Pkey Id                                 |

**SOAP Example**

The following SOAP request deletes the Agent pkey map with id 17585 from Agent 1234. Any other Agent pkey maps associated with Agent 1234 are unaffected.

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    <soap:Header/>
    <soap:Body>
        <res:Update>
            <res:resources>
                <res:Resource>
                    <res:Identity>1234</res:Identity>
                    <res:Type>Agent</res:Type>
                    <res:Fields/>
                    <res:EquipmentMapping>
                        <res:Equipment>
                            <res:Type>Agent Pkey</res:Type>
                            <res:Identity>17585</res:Identity>
                            <res:Status>D</res:Status>
                            <res:Fields/>
                        </res:Equipment>
                    </res:EquipmentMapping>
                </res:Resource>
            </res:resources>
        </res:Update>   </soap:Body>
</soap:Envelope>
```

**Create Agent To Agent Team Membership (Two Remote Equipment Mappings)**

**Description**

This example creates an Agent to Agent Team membership that relates an Agent to an Agent Team where the Agent and the Agent Team are both linked to two different remote equipment instances. In this example, the Agent to Agent Team memberships for both remote equipment instances are specified when the Agent to Agent Team membership is created.
Use Case Sequence

Prerequisites:

- The specified Agent exists and is mapped to at least two remote equipment instances.
- The specified Agent Team exists and is mapped to the same two remote equipment instances.

Table 3-39 Create Agent To Agent Team Membership (Two Remote Equipment Mappings) Use Case

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the Agent record and obtain the Agent Pkey ids for each remote equipment mapping.</td>
</tr>
<tr>
<td>2</td>
<td>Retrieve the Agent Team record and obtain the Agent Pkey ids for each remote equipment mapping.</td>
</tr>
<tr>
<td>3</td>
<td>Create the Agent Agent Team membership, specifying the remote equipment mapping in a pkey as in the SOAP example below.</td>
</tr>
</tbody>
</table>

SOAP Example

The following SOAP request will create a membership between the Agent with id 1234 and pkey maps of 1111 and 2222 and the Agent Team with id 6789 and pkey maps 6666 and 7777.
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:-->  
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>Agent Agent Team Member</res:Type>
          <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T00:00:00</res:EffectiveTo>
          <res:Status>R</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>ParentId</res:Name>
              <res:Value>6789</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>ChildId</res:Name>
              <res:Value>1234</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Supervisor</res:Name>
              <res:Value>false</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>PrimarySupervisor</res:Name>
              <res:Value>false</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>PhysicalMember</res:Name>
              <res:Value>true</res:Value>
            </res:NameValuePair>
          </res:Fields>
          <res:EquipmentMapping>
            <res:Equipment>
              <res:Type>Agent Agent Team Pkey</res:Type>
              <res:Fields>
                <res:NameValuePair>
                  <res:Name>ParentId</res:Name>
                  <res:Value>6666</res:Value>
                </res:NameValuePair>
              </res:Fields>
            </res:Equipment>
          </res:EquipmentMapping>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
Create Precision Attribute

Soap Example

This example creates a Precision Attribute “Mortgages” in the folder with FolderId 00000000-0000-0000-0000-000000000010, where the Attribute is a proficiency attribute (code 4) with a default value of 1.

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:Name>ChildId</res:Name>
      <res:Value>1111</res:Value>
    </res:NameValuePair>
    <res:Name>ClusterResourceId</res:Name>
    <res:Value>{c5ed5103-55cc-420b-8fa9-9d64c2ea339}</res:Value>
    <res:FieldMapping>
      <res:NameValuePair>
        <res:Name>ParentId</res:Name>
        <res:Value>7777</res:Value>
      </res:NameValuePair>
      <res:NameValuePair>
        <res:Name>ChildId</res:Name>
        <res:Value>2222</res:Value>
      </res:NameValuePair>
      <res:NameValuePair>
        <res:Name>ClusterResourceId</res:Name>
        <res:Value>{5103c5ed-d43e-fa4e-98f0-2eaa33a9d64c}</res:Value>
      </res:NameValuePair>
    </res:FieldMapping>
  </soap:Body>
</soap:Envelope>
```
Create Precision Queue

Description

This example creates a Precision Queue, which contains:

- a single Precision Queue Step
- a Precision Queue Step Precision Attribute Member that links the Precision Queue Step to an existing Precision Attribute with id 3333.
The details for the step and the member that links the step and the attribute are specified in the same request as the queue since they must be created at the same time.

When the queue is created, the following items are also created automatically as part of the creation process:
- the specified step
- the specified member that links the step and the attribute
- a Precision Queue Step Precision Queue Member linking the new step with the new queue.

Use Case Sequence

Prerequisites:
- The specified Precision Attribute exists

Table 3-40 Create Precision Queue

<table>
<thead>
<tr>
<th>Action</th>
<th>API Calls and Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create the Precision Queue and at least one related Prevision Queue Step (which in turn must contain at least one Precision Attribute membership), specifying the details for all three items as in the SOAP example below.</td>
</tr>
<tr>
<td>2</td>
<td>Wait for the asynchronous notification if a subscription has been set up (SOAP only), or begin a poll sequence using search or retrieve.</td>
</tr>
</tbody>
</table>

SOAP Example

This SOAP request creates a Precision Queue “Precision Queue 1”, containing one step “Precision Queue 1 Step 1” that links to the Attribute with id “5555”.

The comparison value for this attribute in this step is 7 (field Value1), and the comparison to be used is “greater than or equal to” (specified by a value of 6 in field AttributeRelation).

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <!-- Precision Queue -->
        <res:Resource>
          <!-- Precision Queue 1 -->
          <!-- Precision Queue 1 Step 1 -->
          <!-- Precision Attribute with id 5555 -->
          <!-- AttributeRelation: 6 -->
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
<res:Identity>-1</res:Identity>
<res:Type>IT_PRECISION_QUEUE</res:Type>
<res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
<res:EffectiveTo>2079-06-06T00:00:00</res:EffectiveTo>
<res:Status>R</res:Status>
<res:Fields>
  <res:NameValuePair>
    <res:Name>FolderId</res:Name>
    <res:Value>00000000-0000-0000-0000-000000000010</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>Name</res:Name>
    <res:Value>Precision Queue 1</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>InternalName</res:Name>
    <res:Value>CICM.PQ1</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>AgentOrdering</res:Name>
    <res:Value>1</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>ServiceLevelThreshold</res:Name>
    <res:Value>1</res:Value>
  </res:NameValuePair>
  <res:NameValuePair>
    <res:Name>ServiceLevelType</res:Name>
    <res:Value>1</res:Value>
  </res:NameValuePair>
</res:Fields>

<!-- First Precision Queue Step -->
<res:Identity>-1</res:Identity>
<res:Type>IT_PRECISION_QUEUE_STEP</res:Type>
<res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
<res:EffectiveTo>2079-06-06T00:00:00</res:EffectiveTo>
<res:Status>R</res:Status>
<res:Fields>
  <res:NameValuePair>
    <res:Name>FolderId</res:Name>
    <res:Value>00000000-0000-0000-0000-000000000010</res:Value>
  </res:NameValuePair>
</res:Fields>
**Chapter 3: Resource Management Web Service**

**Examples**

```xml
<res:NameValuePair>
  <res:Name>Name</res:Name>
  <res:Value>Precision Queue 1 Step 1</res:Value>
</res:NameValuePair>
<res:NameValuePair>
  <res:Name>StepOrder</res:Name>
  <res:Value>1</res:Value>
</res:NameValuePair>
<res:NameValuePair>
  <res:Name>WaitTime</res:Name>
  <res:Value>-1</res:Value>
</res:NameValuePair>

<!-- The parent precision queue is being created at the same time as this step -->
<res:NameValuePair>
  <res:Name>PrecisionQueueUrn</res:Name>
  <res:Value>-3</res:Value>
</res:NameValuePair>
</res:Fields>
</res:Resource>

<!-- Precision Queue Step Precision Queue Attribute Member, Comparison is “Attribute id 5555 >= 7” -->
<res:NameValuePair>
  <res:Name>ParentId</res:Name>
  <res:Value>-3</res:Value>
</res:NameValuePair>
<res:NameValuePair>
  <res:Name>ChildId</res:Name>
  <res:Value>5555</res:Value>
</res:NameValuePair>
<res:NameValuePair>
  <res:Name>Value1</res:Name>
  <res:Value>7</res:Value>
</res:NameValuePair>
<res:NameValuePair>
  <res:Name>TermOrder</res:Name>
  <res:Value>1</res:Value>
</res:NameValuePair>
```

**August 2015**

Web Services Reference for Cisco Unified Contact Center Management Portal
Update Precision Queue (Add Precision Queue Step With Two Precision Attributes)

Description

This example adds a Precision Queue Step containing two Precision Attribute expressions to the end of an existing Precision Queue.

The details for the members that link the step and the attributes are specified in the same request as the step since they must be created at the same time.

When the step is created, the member linking the new step with the parent queue is also created automatically as part of the creation process.

After the new step is added, the WaitTime field in the original step must be modified as that step is no longer the last step in the queue. In this example, a ConsiderIf expression is also added to the original step.

This example uses the Save() method since the create and update actions must be done in the same SOAP request (the ConsiderIf could be updated later, but the WaitTime must be updated as the new step is added, because this must be a non-zero value for all steps except the last one).

SOAP Example

This SOAP request creates a new Precision Queue Step “Precision Queue Step 2” in the Precision Queue with id “123456”, that links to the Precision Attributes with id “5555” and “6666”.

The comparison value for attribute “5555” in this step is 4 (field Val ue1 in the step to attribute membership), and the comparison to be used is “greater than or equal to”
(specified by a value of 6 in field AttributeRelation in the step to attribute membership).

The comparison value for attribute “6666” in this step is “true” (field Value1), and the comparison to be used is “equal to” (specified by a value of 1 in field AttributeRelation). The TermRelation field has a value of 2, indicating that this attribute term is to be compared with the previous one using an “OR” operation.

This step is the new last step in the queue, so the existing step, with id “6789” is updated to have a wait time of 20 seconds.

```xml
    <soap:Header/>
    <soap:Body>
        <res:Save>
            <res:resourcesToAdd>
                <res:Resource>
                    <res:Identity>-1</res:Identity>
                    <res:Type>IT_PRECISION_QUEUE_STEP</res:Type>
                    <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
                    <res:EffectiveTo>2079-06-06T00:00:00</res:EffectiveTo>
                    <res:Status>R</res:Status>
                    <res:Fields>
                        <res:NameValuePair>
                            <res:Name>FolderId</res:Name>
                            <res:Value>00000000-0000-0000-0000-000000000010</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>Name</res:Name>
                            <res:Value>Precision Queue 1 Step 2</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>StepOrder</res:Name>
                            <res:Value>2</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>WaitTime</res:Name>
                            <res:Value>-1</res:Value>
                        </res:NameValuePair>
                        <res:NameValuePair>
                            <res:Name>PrecisionQueueUrn</res:Name>
                            <res:Value>123456</res:Value>
                        </res:NameValuePair>
                    </res:Fields>
                </res:Resource>
            </res:resourcesToAdd>
        </res:Save>
    </soap:Body>
</soap:Envelope>
```
Chapter 3: Resource Management Web Service

Examples

<res:Resource>
  <res:Identity>3</res:Identity>
  <res:Type>MT_PRECISION_QUEUE_STEP_PRECISION_ATTRIBUTE_MEMBER</res:Type>
  <res:Fields>
    <res:NameValuePair>
      <res:Name>ParentId</res:Name>
      <res:Value>3</res:Value>
    </res:NameValuePair>
    <res:NameValuePair>
      <res:Name>ChildIdId</res:Name>
      <res:Value>5555</res:Value>
    </res:NameValuePair>
    <res:NameValuePair>
      <res:Name>Value1</res:Name>
      <res:Value>4</res:Value>
    </res:NameValuePair>
    <res:NameValuePair>
      <res:Name>TermOrder</res:Name>
      <res:Value>1</res:Value>
    </res:NameValuePair>
    <res:NameValuePair>
      <res:Name>ParenthesesCount</res:Name>
      <res:Value>0</res:Value>
    </res:NameValuePair>
    <res:NameValuePair>
      <res:Name>TermRelation</res:Name>
      <res:Value>0</res:Value>
    </res:NameValuePair>
    <res:NameValuePair>
      <res:Name>AttributeRelation</res:Name>
      <res:Value>6</res:Value>
    </res:NameValuePair>
  </res:Fields>
</res:Resource>

Member 2,
  
  Comparison is "OR Attribute id 6666 = true" --->
  <res:Resource>
    <res:Identity>3</res:Identity>
    <res:Type>MT_PRECISION_QUEUE_STEP_PRECISION_ATTRIBUTE_MEMBER</res:Type>
  </res:Resource>

轨道上外侧的文本。
<res:Fields>
<res:NameValuePair>
<res:Name>ParentId</res:Name>
<res:Value>-3</res:Value>
</res:NameValuePair>
<res:NameValuePair>
<res:Name>ChildId</res:Name>
<res:Value>6666</res:Value>
</res:NameValuePair>
<res:NameValuePair>
<res:Name>Value1</res:Name>
<res:Value>true</res:Value>
</res:NameValuePair>
<res:NameValuePair>
<res:Name>TermOrder</res:Name>
<res:Value>2</res:Value>
</res:NameValuePair>
<res:NameValuePair>
<res:Name>ParenthesesCount</res:Name>
<res:Value>0</res:Value>
</res:NameValuePair>
<res:NameValuePair>
<res:Name>TermRelation</res:Name>
<res:Value>2</res:Value>
</res:NameValuePair>
<res:NameValuePair>
<res:Name>AttributeRelation</res:Name>
<res:Value>1</res:Value>
</res:NameValuePair>
</res:Fields>
</res:Resource>

---

<res:Resource>
<res:Identity>6789</res:Identity>
<res:Type>IT_PRECISION_QUEUE_STEP</res:Type>
<res:Fields>
<res:NameValuePair>
<res:Name>WaitTime</res:Name>
<res:Value>20</res:Value>
</res:NameValuePair>
</res:Fields>
</res:Resource>

<soap:Body>
</soap:Body>
</soap:Envelope>
Update Precision Queue (Delete Precision Queue Step)

Description

This example deletes the first Precision Queue Step from an existing Precision Queue that contained two Precision Queue Steps.

When the step is deleted, the following items are also deleted automatically:

- the member linking the deleted step with the related attribute
- the member linking the deleted step with the parent queue.

In this example, the StepOrder field of the remaining step needs to be modified as it is now the first step in the queue.

This example uses the `Save()` method since the delete and update actions must be done in the same SOAP request (the StepOrder for the remaining step must be updated as the other step is deleted).

SOAP Example

```xml
  <soap:Header/>
  <soap:Body>
    <res:Save>
      <res:resourcesToDelete>
        <res:Resource>
          <res:Identity>6789</res:Identity>
          <res:Type>IT_PRECISION_QUEUE_STEP</res:Type>
        </res:Resource>
      </res:resourcesToDelete>
      <res:resourcesToAdd>
        <res:Resource>
          <res:Identity>6790</res:Identity>
          <res:Type>IT_PRECISION_QUEUE_STEP</res:Type>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>StepOrder</res:Name>
              <res:Value>1</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resourcesToAdd>
    </res:Update>
  </soap:Body>
</soap:Envelope>
```
4. Subscriptions Web Service

Overview

Description

The Subscriptions Web Service allows the client application to subscribe to receive notifications when there is a change of state to specified items in the Cisco Unified CCE.

When a specified change occurs, the client receives a notification about the change. The notification includes the identity, name and status of the item.

Specification

SOAP

Web/Application server here:

https://<server>:8083/SubscriptionManager?wsdl

where <server> is the name of the Web/Application server.

The Subscriptions Web Service implements the WS-Eventing interface which is specified at http://www.w3.org/Submission/WS-Eventing/.

REST

The Subscriptions Web Services does not support the REST protocol.

Securing Notifications with SSL

Introduction

This section describes how to use SSL to secure and authenticate the communications from the Unified CCMP web server (the server generating the notifications) to the subscriber server (the server running the subscriber client application).

This procedure is optional. It is only necessary if you plan to use the Unified CCMP Subscriptions Web Service, and want to secure and authenticate the communications between the Unified CCMP server and the subscriber server. If you are not using the Unified CCMP Subscriptions Web Service, or if you do not need to secure and authenticate communications from the Unified CCMP server to the subscriber server (for example, if all your servers are co-located in a secure area) then this procedure is not required.
In order to use the Unified CCMP Web Services, you must first secure the Unified CCMP Web Services themselves as described in the *Installation and Configuration Guide for Cisco Unified Contact Center Management Portal*, section *Securing Unified CCMP Web Service APIs with SSL*. This procedure is not optional and you will not be able to use the Unified CCMP Web Services unless you have done this.

When Unified CCMP is first installed, communications between the Unified CCMP server and the subscriber server are secured with a self-signed certificate called *localhost*. This certificate is suitable for a single server in a laboratory environment, but cannot be used to secure a multi-server installation in a production environment.

To secure and authenticate the communications between the Unified CCMP server and the subscriber server:

- obtain and install a suitable digital certificate on the Unified CCMP server (see section *Obtain and Install a Digital Certificate for the Unified CCMP Server*)
- obtain and install a suitable digital certificate on the subscriber server (see section *Obtain and Install a Digital Certificate for the Subscriber Server*)
- configure the Unified CCMP server endpoint behavior to use the two certificates for communication between the servers (see section *Configure the Unified CCMP Server Endpoint Behaviour*)
- install the public key of the subscriber server on the Unified CCMP server (see section *Install the Public Key of the Subscriber Server Certificate onto the Unified CCMP Server*)
- ensure the certificate authority root certificate for the Unified CCMP server certificate is available on the subscriber server (see section *Install the Root CA Certificate on the Subscriber server*)
- restart the services on both servers.

**Obtain and Install a Digital Certificate for the Unified CCMP Server**

This step installs the digital certificate that secures the Unified CCMP side of the communications between the Unified CCMP web server and the subscriber server.

When Unified CCMP was installed, the Unified CCMP Web Services should have been secured with SSL (see *Installation and Configuration Guide for Cisco Unified Contact Center Management Portal*, section *Securing Unified CCMP Web Service APIs with SSL*). If this has not been done, complete that procedure before continuing with the instructions here.

As part of the process of securing the Unified CCMP Web Services with SSL, a digital certificate was obtained and installed on the Unified CCMP server. We recommend that you use the same digital certificate here. If you choose to use the same certificate, then it has already been installed and this step is complete.

If you want to use a different certificate, follow the instructions in *Installation and Configuration Guide for Cisco Unified Contact Center Management Portal* to obtain
and install another certificate on the Unified CCMP server and to grant access to the NETWORK SERVICE user.

**Obtain and Install a Digital Certificate for the Subscriber Server**

This step installs the digital certificate that secures the subscriber server side of the communications between the Unified CCMP web server and the subscriber server.

You may already have a suitable certificate that you can use to secure the subscriber server side of the communications between the Unified CCMP server and the subscriber server. If you do not already have a suitable certificate, consult your platform documentation to find out how to request or generate an external certificate (suitable for public use) or an internal certificate (for secure use within the issuing organization).

When you have obtained the certificate, install it on the subscriber server as described in your platform documentation.

Ensure that the account that will be used to run the subscriber application has full access rights for the certificate. Again, consult your platform documentation if necessary.

**Configure the Unified CCMP Server Endpoint Behaviour**

This step configures the endpoint behavior of the Unified CCMP server to use the server certificates to secure communications between the two servers.

To do this:

1. Identify the *subject distinguished name* of the certificate you have installed on the Unified CCMP server. This is a text string, containing several name-value pairs, that identifies the certificate, for example:

   “CN=gold, OU=dev, O=exony, L=newbury, S=berkshire, C=GB”

   If you do not know the subject distinguished name, you can find it as follows:
   
   a. On the Unified CCMP server, go to the Start menu, and type `mmc` in the command box to open Microsoft Management Console (MMC).

   b. Click File > Add/Remove Snap-in, click Certificates, then Add.

   c. In the Certificates Snap-in dialog box, select Computer Account and click Next.

   d. In the Select Computer dialog box, select Local Computer and click Finish to add the Certificates snap-in to MMC. Click OK.

   e. In MMC, expand the Certificates node and the Personal node, then click Certificates to see the available certificates.

   f. Double-click on the certificate and in the Details tab, select Subject, to see the name-value pairs that make up the subject distinguished name for the
2. Identify the subject distinguished name of the certificate you have installed on the subscriber server. If the subscriber server is a Windows platform you can follow the instructions above. Otherwise, consult your platform documentation to find out how to do this.

3. On the Unified CCMP server, go to **C:\Program Files\Domain Manager\Application Server**, locate the file **Exony.Reporting.Application.Server.exe.config** and open it in a text editor.

   You may want to make a safe copy of this file before you change it.

4. Locate the section that begins

   ```xml
   <behaviour name="ClientCertificateBehavior ">
   and within that section, the section that begins

   ```

5. In the `<clientCredentials>` section, locate the `<clientCertificate>` tag and change the `findValue` attribute to the subject distinguished name of the Unified CCMP server certificate. Leave the other attributes as they are. Using the example above, the contents of the `<clientCertificate>` tag may become:

   ```xml
   <clientCertificate findValue="CN=gold, OU=dev, O=exony, L=newbury, S=berkshire, C=GB"
   storeLocation="LocalMachine" storeName="My"
   x509FindType="FindBySubjectDistinguishedName" />
   ```

6. Also in the `<clientCredentials>` section, locate the `<serviceCertificate>` tag, and below that, the `<defaultCertificate>` tag. Set the `findValue` attribute of the `<defaultCertificate>` tag to the subject distinguished name of the subscriber server certificate. Leave the other attributes as they are. For example, if the subject distinguished name of the certificate was

   "CN=silver, OU=dev, O=exony, L=newbury, S=berkshire, C=GB"

   the contents of the `<defaultCertificate>` tag may become:

   ```xml
   <defaultCertificate findValue="CN=silver, OU=dev, O=exony, L=newbury, S=berkshire, C=GB"
   storeLocation="LocalMachine" storeName="My"
   x509FindType="FindBySubjectDistinguishedName" />
   ```

7. Save the changes to this file.
Install the Public Key of the Subscriber Server Certificate onto the Unified CCMP Server

This step ensures that the Unified CCMP server can encrypt the communications in the way required by the subscriber server.

To install the public key of the subscriber server certificate onto the Unified CCMP server:

1. On the subscriber server, obtain the public key of the subscriber server certificate and save it to a file. Consult your platform documentation to find out how to do this.

2. Copy the file containing the public key to the Unified CCMP server.

3. On the Unified CCMP server, start MMC, expand the Certificates node and the Personal node, then right-click Certificates. Select All Tasks > Import. Follow the steps in the Certificate Import Wizard to import the file containing the subscriber server public key into the Personal certificate store of the local machine.

Install the Root CA Certificate on the Subscriber server

This step confirms the validity of the Unified CCMP web services certificate to the subscriber server. This is done by obtaining the root certificate of the certificate authority (CA) that issued the Unified CCMP web services certificate and installing it on the subscriber server.

This step is only necessary if the provider of the Unified CCMP server certificate is not already trusted by the subscriber server. If the certificate was purchased from one of the major certificate providers, then the root certificate of the CA will almost certainly be pre-loaded on the subscriber server. In this case, the CA is already trusted.

To install the CA root certificate for the Unified CCMP server certificate:

1. Obtain the CA root certificate for the Unified CCMP server certificate. If you did not get this when you purchased the certificate, you can obtain it from the certificate authority. Copy the file containing the CA root certificate to the subscriber server.

2. On the subscriber server, import the CA root certificate into the Trusted Root certificate area. Consult your platform documentation to find out how to do this if necessary.

Restart the Services

1. Save all configuration files and other changes.

2. On the Unified CCMP server, stop and restart the Unified CCMP Web Service.

3. On the subscriber server, stop and restart the application service.
Subscriptions Web Service

This section describes the web service APIs available in the Subscriptions Web Service.

Subscribe

Description

The Subscribe web service API creates a subscription for item or member changes.

Example

Request

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
<s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<wse:Delivery>
<wse:NotifyTo>
<a:Address>https://localhost:33333/Sink</a:Address>
</wse:NotifyTo>
</wse:Delivery>
<wse:Expires>PT15M</wse:Expires>
<wse:Filter xmlns:x="http://www.exony.com/portal">
//x:Id = 1234 and //x:Type = 'Agent'</wse:Filter>
</wse:Subscribe>
</s:Body>
</s:Envelope>
```

In this example, a subscription has been created for Agent items with an ID of 1234.

- The Filter element can contain any valid XPath query which is applied to the incoming status message (described in more detail later).
- The Expires element defines how long the subscription is held by the server. In this example, T15M indicates that the subscription is held for 15 minutes.
- The Delivery element contains the NotifyTo node which holds an Address. This is the SOAP endpoint that implements the IStatus web service interface.
Chapter 4: Subscriptions Web Service

Subscriptions Web Service

Response

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wse-utility-1.0.xsd">
  <s:Header/>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <wse:SubscriptionManager>
        <a:Address>https://localhost:8083/SubscriptionManager</a:Address>
        <a:ReferenceParameters>
          <wse:Identifier>uuid:ecf6f297-6fce-4924-a973-aeb533378ef6</wse:Identifier>
        </a:ReferenceParameters>
      </wse:SubscriptionManager>
    </wse:SubscribeResponse>
  </s:Body>
</s:Envelope>

In the example response, the Identifier is returned which is used in the other web service methods to identify the particular subscription.

The Expires element required has the absolute expiry date time for this subscription.

Get Status

Description

The Get Status web service API allows the client to check the status of a subscription.

Example

Request

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wse-utility-1.0.xsd">
  <s:Header>
  </s:Header>
</s:Envelope>
</s:Body>
</s:Envelope>

Response

  <s:Header>
    <a:RelatesTo>uuid:1128b40a-f9a7-46f0-8ca5-67aa6410607</a:RelatesTo>
    <o:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2009-12-24T10:31:12.187Z</u:Created>
      </u:Timestamp>
    </o:Security>
  </s:Header>
    </wse:GetStatusResponse>
  </s:Body>
</s:Envelope>

Renew

Description

The Renew web service API allows the client to extend the lifetime of a subscription.
Example

Request

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
 xmlns:a="http://www.w3.org/2005/08/addressing"
 xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
 <s:Header>
  <h:Identifier
   xmlns:h="http://schemas.xmlsoap.org/ws/2004/08/eventing"
 </s:Header>
 <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wse:Renew
   <wse:Expires>PT20M</wse:Expires>
  </wse:Renew>
 </s:Body>
</s:Envelope>
```

Response

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
 xmlns:a="http://www.w3.org/2005/08/addressing"
 xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
 <s:Header>
 </s:Header>
 <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wse:RenewResponse
  </wse:RenewResponse>
 </s:Body>
</s:Envelope>
```

Unsubscribe

Description

The Unsubscribe web service API allows the client to cancel an existing subscription.
Example

Request

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <s:Header>
    <h:Identifier xmlns:h="http://schemas.xmlsoap.org/ws/2004/08/eventing"
      uuid:67f4901f-e183-4ea2-83cb-24f07e82d3e9
    </h:Identifier>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  </s:Body>
</s:Envelope>
```

Response

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <s:Header>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:xsd="http://www.w3.org/2001/XMLSchema"/>
</s:Envelope>
```

More Information about Subscriptions and Notifications

Subscribing for Notifications

When the client subscribes for notifications, the client specifies an XPath query as the filter to choose the event messages to receive. The XPath query is applied to this message.

The filter supports the use of the following XPath components:

- `//x:Id`
  The identity of the item to subscribe to

- `//x:Type`
  The item type of items to subscribe to (for example, Agent, Agent Team)
• \(//x:\text{Status}\)
  The status type that the subscription is concerned with. For example a client may be interested in when Agents become ready or if they are deleted.

• \(//x:\text{ContainerId}\)
  The identity of the folder in which items must belong. This may be used to monitor multiple items for a given folder, for example, to send a notification when any agent within a particular folder is updated.

The client can define a subscription query by combining one or more of these XPath components. For example:

• \(//x:\text{Id} = 1234\) and \(//x:\text{Type} = 'Agent'\) will cause a notification to be received by the client when the agent with identity 1234 changes state.

• \(//x:\text{Type} = 'Agent'\) and \(//x:\text{Status} = 'R'\) and \(//x:\text{ContainerId} = '1BAE1951-A9FF-4F17-AC46-FA7605C26569'\) will cause a notification to be sent when any agent in the folder with the identity 1BAE1951-A9FF-4F17-AC46-FA7605C26569 changes status to R.

The Notifications Web Service only sends notifications for items that support status changes. If an API request (for example Create) returns a status of Ready then the status of that item is not updated and no notifications will be sent.

Notification Reporting

Certain resource types require Unified CCMP to remotely provision them and will return a Pending or Synchronize state when fields and memberships are added/updated or deleted. These items should be subscribed to allowing status changes to be obtained as the item moves through the provisioning lifecycle.

An example work flow for the agent creation process is as follows:

1. Agent create request is performed by the client, business logic is applied to the item and it is written to the Unified CCMP Database waiting to be provisioned onto the remote equipment (in this case Unified CCE). Notification is sent out to all subscribers to say that a new agent has been added and its status is synchronizing.

2. The Unified CCMP Provisioning Server picks up the new change and attempt to make the change on Unified CCE.

3. The Unified CCE successfully creates the new agent, the provisioning server then updates the item status in the RDBMS and a new notification is sent to all subscribers that the agent has progressed to Ready status (that is, has been created on Unified CCE).
Not all resource types use the provisioning lifecycle as some type, for example, User, Group etc are not required to be provisioned on any remote equipment. They are added to Unified CCMP with immediate effect. Therefore the notification behavior of these types will be subtly different than those that utilize the provisioning lifecycle detailed above.

The following types of notification are supported by Unified CCMP:

- Support Provisioning Lifecycle – has the behavior described above, these are types that rely on Unified CCMP to provision them on remote equipment. These
types will progress through the provisioning lifecycle and report their progress at each stage using the notifications to report their current state.

- Returns Status on W/S Call - returns an accurate status of the resource where no provisioning lifecycle exists. This method is used for items that are not required to be provisioned on remote equipment for example users/ groups etc. The status is reported at the time of the operation. An error status indicates that the operation was not successful and the resource will remain in its existing state.

- Notification on Create - when a new resource is created a notification request can be expected stating the new item identity and the new item status.

- Notification on Update - when a resource is updated then a notification request can be expected reporting the updated items identity and status.

- Notification on Delete - when a resource is deleted then a notification request can be expected reporting the identity of the resource and its new status.

The following table shows the supported types and the level of notification support that they adopt.

<table>
<thead>
<tr>
<th>Type</th>
<th>Support Provisioning Lifecycle</th>
<th>Returns Status on W/S Call</th>
<th>Notification on Create</th>
<th>Notification on Update</th>
<th>Notification on Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Agent Team</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Call Type</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Directory Number</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Folder</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Group</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>IP Endpoint</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Person</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Skill Group</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>User</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Agent Agent Desktop Member</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Agent Agent Team Member</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
### Handling Notification Failures

Subscription and notification information is not persisted in the event of a failure. Notifications are sent periodically when a status change is identified for any subscribed resource, for example, when AgentA moves from status R to status S. Given the distributed nature of the platform it is possible that an item may progress through more than one state before the notification change is collected.

The following subscription method should therefore be applied to protect client applications when notifications are not received (for example, in the event of a failure).

1. Subscribe to resource x (currently status = R)
2. Perform change (for example, Call Update API, status = S)
3. Await ready notification
4. If notification is not received in 30 seconds call Search for the item identity of resource x
5. If status = R then progress, if status = S then go back to step 3

This technique provides a failsafe tracking mechanism for resources progressing through the provisioning lifecycle.

### API Notification Timeout

**Description**

The asynchronous notifications could be lost, for example, if there are network problems, so if it is critical for a response to be received, then we suggest that the caller implements a timer to retrieve the result.

<table>
<thead>
<tr>
<th>Type</th>
<th>Support Provisioning Lifecycle</th>
<th>Returns Status on W/S Call</th>
<th>Notification on Create</th>
<th>Notification on Update</th>
<th>Notification on Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Skill Group Member</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>IP Endpoint Directory Number Member</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>User Group Member</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Group Group Member</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Sequence Diagram

**Figure 4-1 API Notification Timeout Sequence Diagram**

Closed Loop Poll

**Description**

Moves, adds and changes may be made in the underlying remote equipment. Unified CCMP periodically reads the configuration data from the remote equipment and updates the Unified CCMP data model accordingly. Updating the data model can have two results:

- Resource notifications will be sent to any clients who have subscribed to them.
- Additional Unified CCMP provisioning workflows may be triggered. For example, the addition of a CUCM Line with an IPCC category set in the description field will trigger the addition of a Device Target / Labels per Routing Client on the relevant Unified CCE. And provisioning workflows may trigger additional status changes.
Interaction with Resource Management Web Services

See section Examples for some detailed examples showing how the Subscriptions and Resource Management Web Services interact to complete some common tasks.

Examples

This section contains some detailed examples showing how the Subscriptions and Resource Management Web Services interact to complete some common tasks.

Agent Re-Skilling

A typical agent re-skilling activity may be performed using the following requests and response sequence.
Step 1

Load the agent that is to be re-skilled using the Resource Management Retrieve web service API.

Request

```xml
  <soap:Header/>
  <soap:Body>
    <res:Retrieve>
      <!--Optional:-->
      <res:resourceKeys>
        <!--Zero or more repetitions:-->--
        <res:ResourceKey>
          <res:Identity>17585</res:Identity>
          <res:ResourceType>Agent</res:ResourceType>
        </res:ResourceKey>
      </res:resourceKeys>
    </res:Retrieve>
  </soap:Body>
</soap:Envelope>
```

Response

```xml
  <s:Header>
    <a:RelatesTo>uuid:701c0aa3-12d1-4728-b199-aae95d64514a</a:RelatesTo>
    <s:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2010-01-19T13:57:54.911Z</u:Created>
        <u:Expires>2010-01-19T14:02:54.911Z</u:Expires>
      </u:Timestamp>
    </s:Security>
  </s:Header>
</s:Envelope>
```
<s:Body>
  <RetrieveResponse xmlns="http://www.exony.com/schemas/2009/10/resourcemanagement">
    <RetrieveResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <Resource>
        <Identity>17585</Identity>
        <Type>Agent</Type>
        <EffectiveFrom>2010-01-18T08:58:36</EffectiveFrom>
        <EffectiveTo>2079-06-06T00:00:00</EffectiveTo>
        <Status>R</Status>
        <Changestamp>0</Changestamp>
        <Fields>
          <NameValuePair><Name>FolderId</Name><Value>1bae1951-a9ff-4f17-ac46-fa7605c26569</Value></NameValuePair>
          <NameValuePair><Name>Name</Name><Value>V723.PG1_1.USER67.AG</Value></NameValuePair>
          <NameValuePair><Name>InternalName</Name><Value>V723.PG1_1.USER67.AG</Value></NameValuePair>
          <NameValuePair><Name>Description</Name><Value>User67</Value></NameValuePair>
          <NameValuePair><Name>PeripheralUrn</Name><Value>3529</Value></NameValuePair>
          <NameValuePair><Name>PersonUrn</Name><Value>9409</Value></NameValuePair>
          <NameValuePair><Name>FirstName</Name><Value>User67</Value></NameValuePair>
          <NameValuePair><Name>LastName</Name><Value>User67</Value></NameValuePair>
        </Fields>
      </Resource>
    </RetrieveResult>
  </RetrieveResponse>
</s:Body>
<NameValuePair>
    <Name>PeripheralNumber</Name>
    <Value>50067</Value>
</NameValuePair>
<NameValuePair>
    <Name>PeripheralName</Name>
    <Value/>
</NameValuePair>
<NameValuePair>
    <Name>Supervisor</Name>
    <Value>False</Value>
</NameValuePair>
<NameValuePair>
    <Name>AgentStateTrace</Name>
    <Value>False</Value>
</NameValuePair>
<NameValuePair>
    <Name>DomainName</Name>
    <Value/>
</NameValuePair>
<NameValuePair>
    <Name>DomainLoginName</Name>
    <Value/>
</NameValuePair>
<NameValuePair>
    <Name>DomainUserName</Name>
    <Value/>
</NameValuePair>
<NameValuePair>
    <Name>DomainUserGuid</Name>
    <Value/>
</NameValuePair>
<NameValuePair>
    <Name>DomainPassPhrase</Name>
    <Value/>
</NameValuePair>
<NameValuePair>
    <Name>UserDeletable</Name>
    <Value>True</Value>
</NameValuePair>
<NameValuePair>
    <Name>UserGroupId</Name>
    <Value/>
</NameValuePair>
Step 2

Subscribe to the agent that is to be re-skilled so that a notification is received when their status changes.

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xm xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <wse:Delivery>
        <wse:NotifyTo>
          <a:Address>
            https://localhost:33333/Sink
          </a:Address>
        </wse:NotifyTo>
        <wse:Expires>PT15M</wse:Expires>
        <wse:Filter xmlns:x="http://www.exony.com/portal"/>
      </wse:Delivery>
    </wse:Subscribe>
  </s:Body>
</s:Envelope>
```

Step 3

Load the list of Skill Groups in which the Agent already exists so that they can be displayed for removal and may be excluded from the list of available Skill Groups. The list of available Skill Groups can be retrieved using the Resource Management Search API.
Chapter 4: Subscriptions Web Service

Examples

Request

```xml
  <soap:Header/>
  <soap:Body>
    <res:Search>
      <res:queryString>type:SkillGroup parentof:Agent, 17585</res:queryString>
      <res:excludeFilter/>
    </res:Search>
  </soap:Body>
</soap:Envelope>
```

Replace the selected identity in this request with the identity of the agent retrieved in Step 1.

Response

The response contains a collection of Skill Group resources for the Skill Groups that are mapped to that agent.

```xml
  <s:Header>
    <s:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2010-01-19T14:24:22.545Z</u:Created>
        <u:Expires>2010-01-19T14:29:22.545Z</u:Expires>
      </u:Timestamp>
    </s:Security>
  </s:Header>
  <s:Body>
      <SearchResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Resource>
          <Identity>17321</Identity>
          <Type>SkillGroup</Type>
          <EffectiveFrom>2010-01-18T09:04:51</EffectiveFrom>
          <EffectiveTo>2079-06-06T00:00</EffectiveTo>
          <Status>R</Status>
        </Resource>
      </SearchResult>
    </SearchResponse>
  </s:Body>
</s:Envelope>
```
<Changestamp>0</Changestamp>
<Fields>
  <NameValuePair>
    <Name>MediaRoutingDomainUrn</Name>
    <Value>8377</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>PeripheralUrn</Name>
    <Value>3529</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>FolderId</Name>
    <Value>1bae1951-a9ff-4f17-ac46-fa7605c26569</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>Name</Name>
    <Value></Value>
  </NameValuePair>
  <NameValuePair>
    <Name>InternalName</Name>
    <Value>AvayaPG_1.Cisco_Voice.Avaya2.pri</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>Description</Name>
    <Value></Value>
  </NameValuePair>
  <NameValuePair>
    <Name>PeripheralNumber</Name>
    <Value>2</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>PeripheralName</Name>
    <Value>AvayaSkill2.pri</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>AvailableHoldoffDelay</Name>
    <Value>-1</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>Priority</Name>
    <Value>1</Value>
  </NameValuePair>
  <NameValuePair>
    <Name>Extension</Name>
    <Value></Value>
  </NameValuePair>
  <NameValuePair>
    <Name>IPTA</Name>
    <Value></Value>
  </NameValuePair>
</Fields>
<NameValuePair><Name>ServiceLevelThreshold</Name><Value>0</Value></NameValuePair><NameValuePair><Name>ServiceLevelType</Name><Value>0</Value></NameValuePair><NameValuePair><Name>DefaultEntry</Name><Value>0</Value></NameValuePair><NameValuePair><Name>SubSkillGroupMask</Name><Value/></NameValuePair><NameValuePair><Name>UserDeletable</Name><Value>True</Value></NameValuePair></Fields></Resource><Resource><Identity>17329</Identity><Type>SkillGroup</Type><EffectiveFrom>2010-01-18T09:04:51</EffectiveFrom><EffectiveTo>2079-06-06T00:00:00</EffectiveTo><Status>R</Status><Changestamp>0</Changestamp><Fields><NameValuePair><Name>MediaRoutingDomainUrn</Name><Value>8377</Value></NameValuePair><NameValuePair><Name>PeripheralUrn</Name><Value>3529</Value></NameValuePair><NameValuePair><Name>FolderId</Name><Value>1bae1951-a9ff-4f17-ac46-fa7605c26569</Value></NameValuePair><NameValuePair><Name>Name</Name><Value/></NameValuePair></Fields></Resource>
<NameValuePair>
   <Name>InternalName</Name>
</NameValuePair>
<NameValuePair>
   <Name>Description</Name>
   <Value></Value>
</NameValuePair>
<NameValuePair>
   <Name>PeripheralNumber</Name>
   <Value>3</Value>
</NameValuePair>
<NameValuePair>
   <Name>PeripheralName</Name>
   <Value>AvayaSkill3.pri</Value>
</NameValuePair>
<NameValuePair>
   <Name>AvailableHoldoffDelay</Name>
   <Value>-1</Value>
</NameValuePair>
<NameValuePair>
   <Name>Priority</Name>
   <Value>1</Value>
</NameValuePair>
<NameValuePair>
   <Name>Extension</Name>
   <Value></Value>
</NameValuePair>
<NameValuePair>
   <Name>IPTA</Name>
   <Value>N</Value>
</NameValuePair>
<NameValuePair>
   <Name>ServiceLevelThreshold</Name>
   <Value>0</Value>
</NameValuePair>
<NameValuePair>
   <Name>ServiceLevelType</Name>
   <Value>0</Value>
</NameValuePair>
<NameValuePair>
   <Name>DefaultEntry</Name>
   <Value>0</Value>
</NameValuePair>
<NameValuePair>
   <Name>SubSkillGroupMask</Name>
   <Value></Value>
</NameValuePair>
This agent is assigned to 2 skill groups. The highlighted skill group identities are used in the next step to exclude them from the list of available selections.

**Step 4**

Load the list of Skill Groups that the agent can be added to using the Resource Management Search web service API. The Skill Groups that have already been mapped are excluded from the results so that they are not available for selection.

The Resource Management Search API is used to retrieve all Skill Groups on a Peripheral. This uses the Peripheral Urn from the agent retrieved in step 1 (agents may only be skilled to skill groups on the same peripheral).

**Request**

```
<?xml version="1.0" encoding="UTF-8"?>
  <soap:Header/>
  <soap:Body>
    <res:Search>
      <res:queryString>type:SkillGroup childof:Peripheral,3529</res:queryString>
      <res:excludeFilter>
        <res:ResourceKey>
          <res:Identity>17321</res:Identity>
        </res:ResourceKey>
        <res:ResourceType>SkillGroup</res:ResourceType>
      </res:excludeFilter>
      <res:excludeFilter>
        <res:ResourceKey>
          <res:Identity>17329</res:Identity>
        </res:ResourceKey>
        <res:ResourceType>SkillGroup</res:ResourceType>
      </res:excludeFilter>
    </res:Search>
  </soap:Body>
</soap:Envelope>
```
Response

The response from this request returns all the skill groups on the peripheral, except the skill groups with identities 17321 and 17329.

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <s:Header>
    <a:Action s:mustUnderstand="1">
    </a:Action>

    <o:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2010-01-19T14:40:09.175Z</u:Created>
        <u:Expires>2010-01-19T14:45:09.175Z</u:Expires>
      </u:Timestamp>
    </o:Security>
  </s:Header>
</s:Envelope>
```
<s:Body>
 <SearchResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Resource>
   <Identity>17241</Identity>
   <Type>SkillGroup</Type>
   <EffectiveFrom>2010-01-18T09:04:49</EffectiveFrom>
   <EffectiveTo>2079-06-06T00:00:00</EffectiveTo>
   <Status>R</Status>
   <Changestamp>0</Changestamp>
   <Fields>
    <NameValuePair>
     <Name>MediaRoutingDomainUrn</Name>
     <Value>8377</Value>
    </NameValuePair>
    <NameValuePair>
     <Name>PeripheralUrn</Name>
     <Value>3529</Value>
    </NameValuePair>
    <NameValuePair>
     <Name>FolderId</Name>
     <Value>1bae1951-a9ff-4f17-ac46-fa7605c26569</Value>
    </NameValuePair>
    <NameValuePair>
     <Name>Name</Name>
     <Value></Value>
    </NameValuePair>
    <NameValuePair>
     <Name>InternalName</Name>
     <Value>PG1_1.Cisco_Voice.default.72576</Value>
    </NameValuePair>
    <NameValuePair>
     <Name>Description</Name>
     <Value></Value>
    </NameValuePair>
    <NameValuePair>
     <Name>PeripheralNumber</Name>
     <Value>19882</Value>
    </NameValuePair>
    <NameValuePair>
     <Name>PeripheralName</Name>
     <Value>000474608774</Value>
    </NameValuePair>
   </Fields>
  </Resource>
 </SearchResult>
</SearchResponse>
<NameValuePair>
  <Name>AvailableHoldoffDelay</Name>
  <Value>-1</Value>
</NameValuePair>
<NameValuePair>
  <Name>Priority</Name>
  <Value>0</Value>
</NameValuePair>
<NameValuePair>
  <Name>Extension</Name>
  <Value/>
</NameValuePair>
<NameValuePair>
  <Name>IPTA</Name>
  <Value>Y</Value>
</NameValuePair>
<NameValuePair>
  <Name>ServiceLevelThreshold</Name>
  <Value>0</Value>
</NameValuePair>
<NameValuePair>
  <Name>ServiceLevelType</Name>
  <Value>0</Value>
</NameValuePair>
<NameValuePair>
  <Name>DefaultEntry</Name>
  <Value>1</Value>
</NameValuePair>
<NameValuePair>
  <Name>SubSkillGroupMask</Name>
  <Value/>
</NameValuePair>
<NameValuePair>
  <Name>UserDeletable</Name>
  <Value>True</Value>
</NameValuePair>
</Fields>
</Resource>
<Resource>
  <Identity>17249</Identity>
  <Type>SkillGroup</Type>
  <EffectiveFrom>2010-01-18T09:04:49</EffectiveFrom>
  <EffectiveTo>2079-06-06T00:00:00</EffectiveTo>
  <Status>R</Status>
  <Changestamp>0</Changestamp>
</Resource>
<Fields>
<NameValuePair>
<Name>MediaRoutingDomainUrn</Name>
<Value>8377</Value>
</NameValuePair>
<NameValuePair>
<Name>PeripheralUrn</Name>
<Value>3529</Value>
</NameValuePair>
<NameValuePair>
<Name>FolderId</Name>
<Value>1bae1951-a9ff-4f17-ac46-fa7605c26569</Value>
</NameValuePair>
<NameValuePair>
<Name>Name</Name>
<Value></Value>
</NameValuePair>
<NameValuePair>
<Name>InternalName</Name>
<Value>Skill2</Value>
</NameValuePair>
<NameValuePair>
<Name>Description</Name>
<Value></Value>
</NameValuePair>
<NameValuePair>
<Name>PeripheralNumber</Name>
<Value>1</Value>
</NameValuePair>
<NameValuePair>
<Name>PeripheralName</Name>
<Value>Skill2</Value>
</NameValuePair>
<NameValuePair>
<Name>AvailableHoldoffDelay</Name>
<Value>-1</Value>
</NameValuePair>
<NameValuePair>
<Name>Priority</Name>
<Value>0</Value>
</NameValuePair>
<NameValuePair>
<Name>Extension</Name>
<Value></Value>
</NameValuePair>
</Fields>
Add the Agent to a Skill Group. The Resource Management Create web service API is called passing the identity of the Agent and Skill Group to create a new Agent Skill Group Member.

**Request**

Here is the agent skill group member creation request. ParentItemUrn describes the identity of the skill group to which the agent is being associated. ChildItemUrn describes the identity of the agent that is being added to the skill group.
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>Agent Skill Group Member</res:Type>
          <res:EffectiveFrom>1900-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T23:59:59.9999999</res:EffectiveTo>
          <res:Status>S</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>ParentItemUrn</res:Name>
              <res:Value>17249</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>ChildItemUrn</res:Name>
              <res:Value>17585</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>

**Response**

The identity in the response is the identity of the Agent Skill Group Member object that is being removed.
Step 6

Notification received that the Agent has returned to ready (R) status.

Notification of an agent change is retrieved first when the re-skilling change is committed informing that the status has changed to Synchronizing (S) and then again once the change has been committed and the item has progressed to Ready (R) status.
Step 7

Delete the Agent from a Skill Group

Once the agent returns to Ready (R) status the Resource Management Delete web service API is called passing the identity of the Agent Skill Group Member that is to be removed.

To delete a Agent from a Skill Group we need to obtain the identity of the Agent Skill Group Member object that maps the two items together.

This can be achieved using the following search query in the Resource Management Search API to retrieve all of the Agent Skill Group Member objects for skill groups mapped to this agent:

```
memberbychild:Agent,17585,"Agent Skill Group Member"
```

The identity of the Agent Skill Group Member that is to be removed should then be used in the Resource Management Delete web service API. The request for this API will look like this:
Request

```xml
  <soap:Header/>
  <soap:Body>
    <res:Delete>
      <res:resourceKeys>
        <res:ResourceKey>
          <res:Identity>14757</res:Identity>
          <res:ResourceType>Agent SkillGroup Member</res:ResourceType>
        </res:ResourceKey>
      </res:resourceKeys>
    </res:Delete>
  </soap:Body>
</soap:Envelope>
```

This will produce a response similar to this:

Response

```xml
  <s:Header>
    <a:RelatesTo>uuid:c4a43490-1bd8-4856-8bbf-8516f56a1904</a:RelatesTo>
    <a:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2010-01-19T15:32:58.487Z</u:Created>
      </u:Timestamp>
    </a:Security>
  </s:Header>
</s:Envelope>
```
Step 8

Notification received that the Agent has returned to ready (R) status.

Notification of an agent change is retrieved first when the re-skilling change is committed informing that the status has changed to Synchronizing (S) and then again once the change has been committed and the item has progressed to Ready (R) status.

**Request**

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header></s:Header>
  <s:Body>
      <status xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Id>17585</Id>
        <Status>R</Status>
        <Type>Agent</Type>
        <ContainerId>1bae1951-a9ff-4f17-ac46-fa7605c26569</ContainerId>
      </status>
    </OnStatusChange>
  </s:Body>
</s:Envelope>
```

**Agent Creation**

The following steps must be performed for a typical agent creation activity:

A typical agent creation activity may be performed using the following requests and response sequence.
Step 1

Create a Person.

Create a person record using the Resource Management Create web service. The XML request for this will look similar to this.

**Request**

```xml
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <!--Optional:-->
      <res:resources>
        <!--Zero or more repetitions:-->  
        <res:Resource>
          <res:Identity>-1</res:Identity>
          <res:Type>Person</res:Type>
          <res:EffectiveFrom>0001-01-01T00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06T23:59:59.9999999</res:EffectiveTo>
          <res:Status>S</res:Status>
          <res:Changestamp>0</res:Changestamp>
          <res:Fields>
            <!--Zero or more repetitions:-->  
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>221C6722-B830-4848-9521-35B2DD8757D7</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Name</res:Name>
              <res:Value>JohnSmith</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>FirstName</res:Name>
              <res:Value>John</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>LastName</res:Name>
              <res:Value>Smith</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Description</res:Name>
              <res:Value>A description</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
This request will return the identity of the new person record which we will need later on. In this example the identity of the person returned is 21365. This is returned from the web service in the following XML structure:
Response

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
 xmlns:a="http://www.w3.org/2005/08/addressing"
 xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
 wssecurity-utility-1.0.xsd">
  <s:Header>
    <a:Action
      s:mustUnderstand="1">
      http://www.exony.com/schemas/2009/10/resourceman-
      agement/IResourceManagement/CreateResponse</a:Action>
    <o:Security s:mustUnderstand="1"
      xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
      wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2010-01-19T11:29:22.429Z</u:Created>
        <u:Expires>2010-01-19T11:34:22.429Z</u:Expires>
      </u:Timestamp>
    </o:Security>
  </s:Header>
  <s:Body>
    <CreateResponse
      xmlns="http://www.exony.com/schemas/2009/10/resourceman-
      agement">
      <CreateResult
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <RequestResult>
          <Identity>21365</Identity>
          <Status>S</Status>
          <Errors i:nil="true"/>
        </RequestResult>
      </CreateResult>
    </CreateResponse>
  </s:Body>
</s:Envelope>

Step 2

Subscribe to the Person.

After creation the returned identity should immediately be subscribed to so that when
the status is updated to ready, the associated agent may be created.
The subscription request must be made to the Subscribe method using an XML
request as follows:
Chapter 4: Subscriptions Web Service

Subscriptions Web Service Examples

Web Services Reference for Cisco Unified Contact Center Management Portal
August 2015

---

**Request**

```
<: Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
 xmlns:a="http://www.w3.org/2005/08/addressing"
 xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
 <: Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <wse:Delivery>
 <wse:NotifyTo>
 <a:Address>https://localhost:33333/Sink</a:Address>
 </wse:NotifyTo>
 </wse:Delivery>
 <wse:Expires>PT15M</wse:Expires>
 <wse:Filter xmlns:x="http://www.exony.com/portal">//x:Id = 21365 and //x:Type = 'Person'</wse:Filter>
 </wse:Subscribe>
 </: Body>
</: Envelope>
```

The Address element must be updated to an endpoint that implements the IStatus web service interface and the filter must include the person identity and type from the above request. The expiry of this notification is set to 15 minutes.

It is possible that the item may have changed status between the create and subscription request. Therefore it is recommended that the Resource Management Search API is called after the subscription is setup to ensure that the status has not already changed from synchronizing (S).

---

**Note**

---

**Step 3**

Receive the Person subscription notification.

Within the 15 minute subscription period a notification will be sent to the subscribed endpoint when the associated item has changed. This notification will be an XML request in the following format:

**Request**

```
<: Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
 <: Header>
 <: Body>
 <status xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
 <status xmlns:s="http://www.exony.com/schemas/2009/10/status">
 <status xmlns:s="http://www.w3.org/2001/XMLSchema-instance">
```

---

Web Services Reference for Cisco Unified Contact Center Management Portal
August 2015

283
Step 4

Create the Agent.

Once the notification request has been fired and the created person has moved into the ready status (R) then the agent creation request can be sent to the Resource Management Create web service.

The following XML request creates the agent associating the newly created person:

**Request**

```
  <soap:Header/>
  <soap:Body>
    <res:Create>
      <res:resources>
        <!--Zero or more repetitions:-->  
        <res:Resource>
          <res:Type>Agent</res:Type>
          <res:EffectiveFrom>2009-01-01 00:00:00</res:EffectiveFrom>
          <res:EffectiveTo>2079-06-06 00:00:00</res:EffectiveTo>
          <res:Status>S</res:Status>
          <res:Fields>
            <res:NameValuePair>
              <res:Name>FolderId</res:Name>
              <res:Value>9F44B644-7C24-40CD-9DB9-6D1175DD7FE3</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>PeripheralUrn</res:Name>
              <res:Value>-1</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>PersonUrn</res:Name>
              <res:Value>21365</res:Value>
            </res:NameValuePair>
            <res:NameValuePair>
              <res:Name>Name</res:Name>
              <res:Value>21365</res:Value>
            </res:NameValuePair>
          </res:Fields>
        </res:Resource>
      </res:resources>
    </res:Create>
  </soap:Body>
</soap:Envelope>
```
<res:NameValuePair><res:Name>Agent1</res:Name><res:Value>Jim</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>LastName</res:Name><res:Value>Smith</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>PeripheralNumber</res:Name><res:Value>132456</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>PeripheralName</res:Name><res:Value>Agent1</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>Supervisor</res:Name><res:Value>True</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>DomainName</res:Name><res:Value>DOMAIN</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>DomainLoginName</res:Name><res:Value>DOMAIN\Agent1</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>DomainUserName</res:Name><res:Value>Agent1</res:Value></res:NameValuePair>
<res:NameValuePair><res:Name>DomainUserGuid</res:Value>66965475b1d6a448aafc8fb6deac2b2f</res:NameValuePair>
The response for the above request will return the identity of the new agent.

Response

```xml
  <soap:Header>
    <o:Security s:mustUnderstand="1">
      <u:Timestamp u:Id="_0">
        <u:Created>2010-01-19T11:29:22.429Z</u:Created>
        <u:Expires>2010-01-19T11:34:22.429Z</u:Expires>
      </u:Timestamp>
    </o:Security>
  </soap:Header>
  <soap:Body>
      <CreateResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <RequestResult>
          <Identity>21369</Identity><Status>S</Status><Errors i:nil="true"/>
        </RequestResult>
      </CreateResult>
    </CreateResponse>
  </soap:Body>
</soap:Envelope>
```

Step 5

Subscribe to the Agent using the new identity.

Subscribe to the agent so that a notification is sent when the agent is updated. This will allow the end user to be informed once the change has been made to the remote system. The following XML request should be sent to the Subscriptions web service subscribe API:
Step 6

Receive the Agent Subscription Notification.

Within the 15 minute subscription period a notification will be sent to the subscribed endpoint when the associated item has changed. This notification will be an XML request in the following format:

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
    <s:Header></s:Header>
    <s:Body>
            <status xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
                <Id>21369</Id>
                <Status>R</Status>
                <Type>Agent</Type>
                <ContainerId>9F44B644-7C24-40CD-9DB9-6D1175DD7FE3</ContainerId>
            </status>
            <Cont ail id="D9F44B644-7C24-40CD-9DB9-6D1175DD7FE3">
                <Cont ail id="D9F44B644-7C24-40CD-9DB9-6D1175DD7FE3">
        </s:Body>
</s:Envelope>
```

Once this notification is received then the agent has been successfully created and may be skilled or added to an agent team.
Whilst these examples demonstrate a solution for a single item, it may be considered more efficient to setup and maintain a recurring subscription to a given type. For example, if the Subscribe API is called specifying just a type and not an identity then notifications will be sent for all changes to all items of that type.

### Dialed Number From /Unallocated folder

Resources that are not allocated to a tenant or shared across multiple tenants are held in the equipment specific sub folder under the /Unallocated folder. A typical example of such a resource is the Dialed Number resource type.

To using the Resource Management Web Services to obtain a list of unallocated Dialed Numbers:

**Step 1**

Use the Search API to obtain the identity of the folder containing unallocated Dialed Numbers for this Unified CCE:

**Request**

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  <soap:Header/>
  <soap:Body>
    <res:Search>
      <res:queryString>
        type:Folder folder:"00000002-0000-0000-0000-000000000005"
        name:ProvisioningCICM</res:queryString>
    </res:Search>
  </soap:Body>
</soap:Envelope>
```

The identity of the unallocated folder used in the above search query is fixed for all Unified CCMP installations.
Response

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
    xmlns:a="http://www.w3.org/2005/08/addressing"
    xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
    <s:Header>
        <a:Action
    </s:Header>
    <s:Body>
        <SearchResponse
            xmlns="http://www.exony.com/schemas/2009/10/resourcemanagement">
            <SearchResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
                <Resource>
                    <Identity>dc68cb53-4654-49aa-a947-24592fd1bc39</Identity>
                    <Type>Folder</Type>
                    <EffectiveFrom>0001-01-01T00:00:00</EffectiveFrom>
                    <EffectiveTo>2079-06-06T23:59:59.9999999</EffectiveTo>
                    <Status>R</Status>
                    <Changestamp>0</Changestamp>
                    <Fields>
                        <NameValuePair>
                            <Name>FolderId</Name>
                            <Value>00000002-0000-0000-0000-000000000005</Value>
                        </NameValuePair>
                        <NameValuePair>
                            <Name>Name</Name>
                            <Value>ProvisioningCICM</Value>
                        </NameValuePair>
                    </Fields>
                </Resource>
            </SearchResult>
        </SearchResponse>
    </s:Body>
</s:Envelope>
```
Chapter 4: Subscriptions Web Service

Examples

Record the identity of the returned resource as this is used in the next query to retrieve the list of unallocated Dialed Numbers.

Step 2

Call the Search API using the retrieved Folder ID to request the list of unallocated Dialed numbers for this Unified CCE.

Request

```
  <soap:Header/>
  <soap:Body>
    <res:Search>
      <res:queryString type="Dialed Number" folder:"
[d68cb53-4654-49aa-a947-24592f1d1bc39" latest:1 status:R max:10"/>
    </res:Search>
  </soap:Body>
</soap:Envelope>
```
The list of Dialed Numbers contained within the unallocated folder is returned to the client:

**Response**

```xml
<Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope
xmlns:a="http://www.w3.org/2005/08/addressing
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wseecurity-utility-1.0.xsd">
  <Header>
    <Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wseecurity-secext-1.0.xsd">
      <Timestamp u:Id="_0">
        <Created>2010-03-12T16:59:02.937Z</Created>
        <Expires>2010-03-12T17:04:02.937Z</Expires>
      </Timestamp>
    </Security>
  </Header>
  <Body>
      <SearchResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Resource>
          <Identity>31081</Identity>
          <Type>Dialed Number</Type>
          <EffectiveFrom>2010-03-12T16:32:46.857</EffectiveFrom>
          <EffectiveTo>2079-06-06T00:00:00</EffectiveTo>
          <Status/>
          <Changestamp>0</Changestamp>
          <Fields>
            <NameValuePair>
              <Name>FolderId</Name>
              <Value>dc68cb53-4654-49aa-a947-24592f1b3c39</Value>
            </NameValuePair>
            <NameValuePair>
              <Name>Name</Name>
              <Value/> <!-- InternalName: V723.JONO.DN -->
            </NameValuePair>
            <NameValuePair>
              <Name>Description</Name>
              <Value/> <!-- Description: see above -->
            </NameValuePair>
          </Fields>
        </Resource>
      </SearchResult>
    </SearchResponse>
  </Body>
</Envelope>
```

24592f1b3c39 Va...
```xml
<NameValuePair>
  <Name>Digits</Name>
  <Value>456</Value>
</NameValuePair>
<NameValuePair>
  <Name>InternalName</Name>
  <Value>V723.FG.DN</Value>
</NameValuePair>
<NameValuePair>
  <Name>Description</Name>
  <Value/></Value>
</NameValuePair>
<NameValuePair>
  <Name>PermitApplicationRouting</Name>
  <Value>False</Value>
</NameValuePair>
</Fields>
</Resource>
<Resource>
  <Identity>31089</Identity>
  <Type>Dialed Number</Type>
  <EffectiveFrom>2010-03-12T16:32:46.96</EffectiveFrom>
  <EffectiveTo>2079-06-06T00:00:00</EffectiveTo>
  <Status/>
  <Changestamp>0</Changestamp>
  <Fields>
    <NameValuePair>
      <Name>FolderId</Name>
      <Value>dc68cb53-4654-49aa-a947-24592f d1bc39</Value>
    </NameValuePair>
    <NameValuePair>
      <Name>Name</Name>
      <Value/></Value>
    </NameValuePair>
    <NameValuePair>
      <Name>InternalName</Name>
      <Value>V723.FG.DN</Value>
    </NameValuePair>
    <NameValuePair>
      <Name>Description</Name>
      <Value/></Value>
    </NameValuePair>
    <NameValuePair>
      <Name>PermitApplicationRouting</Name>
      <Value>False</Value>
    </NameValuePair>
  </Fields>
</Resource>
```
Chapter 4: Subscriptions Web Service

Examples

<EffectiveFrom>2010-03-12T16:32:46.963</EffectiveFrom>
<EffectiveTo>2079-06-06T00:00:00</EffectiveTo>
<Status/>
<br />
<Fields>
<NameValuePair>
<Name>FolderId</Name>
<Value>dc68cb53-4654-49aa-a947-24592fd1bc39</Value>
</NameValuePair>
<NameValuePair>
<Name>Name</Name>
<Value/></Value>
</NameValuePair>
<NameValuePair>
<Name>InternalName</Name>
<Value>V723.DN45.DN</Value>
</NameValuePair>
<NameValuePair>
<Name>Description</Name>
<Value/></Value>
</NameValuePair>
<NameValuePair>
<Name>Digits</Name>
<Value>436436745</Value>
</NameValuePair>
<NameValuePair>
<Name>PermitApplicationRouting</Name>
<Value>False</Value>
</NameValuePair>
</Fields>
</Resource>
</SearchResult>
</SearchResponse>
</s:Body>
</s:Envelope>

When returning large sets of resources the search engine Max and Offset keywords should be used to return pages of resources.

**Step 3**

Once the list is obtained then unallocated Dialed Numbers should be moved to the associated tenant folder when they are to be allocated. This is achieved using the Update API, passing in the details of the item to be moved and the associated folder ids as follows:
Request

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
               xmlns="http://www.exony.com/schemas/2009/10/resourcemanagement">
  <soap:Header/>
  <soap:Body>
    <Update>
      <resources>
        <Resource>
          <Identity>31089</Identity>
          <Type>Dialed Number</Type>
          <EffectiveFrom>2010-03-11T12:19:37</EffectiveFrom>
          <EffectiveTo>2011-03-12T16:32:45.96</EffectiveTo>
          <Status>R</Status>
          <Changestamp>0</Changestamp>
          <Fields>
            <NameValuePair>
              <Name>FolderId</Name>
              <Value>DC63CB53-4654-49AA-A947-24592FD1BC39</Value>
            </NameValuePair>
            <NameValuePair>
              <Name>NewFolderId</Name>
              <Value>240253C4-3F30-48C5-A3E6-65B3D94B100E</Value>
            </NameValuePair>
            <NameValuePair>
              <Name>Name</Name>
              <Value>V723.DN45.DN</Value>
            </NameValuePair>
            <NameValuePair>
              <Name>InternalName</Name>
              <Value>V723.DN45.DN</Value>
            </NameValuePair>
            <NameValuePair>
              <Name>Description</Name>
              <Value/>
            </NameValuePair>
            <NameValuePair>
              <Name>RoutingClientUrn</Name>
              <Value>22049</Value>
            </NameValuePair>
            <NameValuePair>
              <Name>MediaRoutingDomainUrn</Name>
              <Value>10789</Value>
            </NameValuePair>
            <NameValuePair>
              <Name>Digits</Name>
              <Value>436436745</Value>
            </NameValuePair>
          </Fields>
        </Resource>
      </resources>
    </Update>
  </soap:Body>
</soap:Envelope>
```
This moves the dialed number from the folder DC68CB53-4654-49AA-A947-24592FD1BC39 to the folder 240253C4-3F30-48C5-A3E6-65B3D94B100E.

When moving items between tenant folders or to or from the unallocated folder, Unified CCMP may delete and recreate the item to enforce security on any associated historical data. This means that the identity for the item will change, and so should be updated in the remote system for future web service calls. The response of the Update request contains the identity of all updated items and allows remote systems storing these keys to be updated.

An example response for the above request where a delete and recreate has been performed is as follows:

**Response**

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <soap:Header>
    <o:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2010-03-16T12:31:33.991Z</u:Created>
        <u:Expires>2010-03-16T12:36:33.991Z</u:Expires>
      </u:Timestamp>
    </o:Security>
  </soap:Header>
  <soap:Body>
    <UpdateResponse xmlns="http://www.exony.com/schemas/2009/10/resourcemanagement">
      <UpdateResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <RequestResult/>
      </UpdateResult>
    </UpdateResponse>
  </soap:Body>
</soap:Envelope>
```
Note the new identity and use it for all future web service calls relating to this resource.
Appendix A. CCMP Technical Overview

Web Service API Architecture

The Unified CCMP Web Services API implementation exposes a number of web services allowing remote provisioning requests to be serviced for third party applications.

These third party applications are typically B2B applications rather than composite client/server interactions.

This first of these services which is identified as the "Resource Management" service exposes a web service layer on top of the existing platforms provisioning abstraction layer. Remote requests are serviced via the web service, building low level objects from passed in type-safe parameters. Provisioning changes are then executing using the existing platform provisioning mechanisms committing change to the Cisco Unified CCE/ Cisco Unified CM environment using their exposed software interfaces. For this service Unified CCMP supports two web service protocols, REST and SOAP for maximum interoperability.

The next of these services which is identified as the “Subscription Management” service exposes another web service layer to allow third party applications the ability to subscribe for events when a given resource is changed. This subscription based event service can be used by third party applications to monitor resource items progress through the provisioning life cycle. The REST protocol cannot be used with this web service.

Web Service API Design Concepts

The Unified CCMP Web Services are hosted within the Unified CCMP application server and do not use the Web Server components at all. Accordingly, they are stateless with a mid-tier cache and distributed for maximum scaling through load balancers. Concurrency checks and two-phase atomic commits (2PC) are implemented in the underlying transactional RDBMS layer for any provisioning write request.

Clients must be aware that each consecutive request can be load balanced to a different server and that the usual race conditions will be present as data makes its way through the systems. For example; a Create call will return an identifier from a Side A server but a subsequent Search/Retrieve on Side B may not see that identifier for up to 10 seconds as the mid-tier caches are updated. Note: this is not a problem in practice since the client already has the new identifier and it can therefore be used in a subsequent provisioning request since write requests are always re-loaded from the RDBMS irrespective of the cache state.
Performance Tips

Like all secure web service stacks, the excellent interoperability offered by this technology comes at a high performance cost in terms of throughput and latency. There are many reasons for this, including HTTP bloat and network latency.

Some common observations and tips when using the Unified CCMP Web Services:

- Web service latencies are measured in the hundreds to thousands of milliseconds which is significantly slower than the binary streaming operations between App Servers to Databases found in traditional client server systems. Hence, make each call count by requesting or doing as much as possible in each call.

- HTTP/S is a best-effort delivery service. This means any request could simply be dropped. Hence, the clients of the Unified CCMP Web Services have to handle this and retry.

- Take the frequency of the messaging into account. Remedy: For provisioning, the majority of data interactions are read-many/write-seldom making them suitable for caching techniques in the higher levels; ideally caching at the edge nearest the users. In the context of many users making many similar small calls then this can be simply implemented as taking the results out of a higher level cache such as memcache.

- High volume aggregation services which involve many elements of state based data queries can be particularly difficult to scale. Hence, replicate key data items into a high level persistent model (the Master Data Source) and use regular re-synchronization, ideally retrieving data during off-hours in large, course-grained transactions.

- Repeated client calls to access server state can choke a network and degrade the server performance. So cache data on the client whenever possible to avoid requests to the server.

- The SOAP protocol is more resource-intensive than the REST equivalent. SOAP has a high intrinsic overhead because it has to extract the SOAP envelope, parse the contained XML information and handle the typing information required in every SOAP message. The XML data cannot be optimized very much, and the binary representation of an object may be expanded by around 500% when encoded in XML.
Appendix B. SOAP Messages

SOAP Request Format

Requests sent to the Unified CCMP Web Services are in the standard SOAP format. The standard SOAP envelope XML format is shown below.

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <!-- Message contents here -->
  </soap:Body>
</soap:Envelope>
```

SOAP Response Format

Responses from the Unified CCMP Web Services are in the standard SOAP format. The standard SOAP envelope XML format is shown below.

```xml
<SOAP-Envelope xmlns="http://www.w3.org/2003/05/soap-envelope" xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <!-- Message contents here -->
  </soap:Body>
</SOAP-Envelope>
```
Appendix C. Valid Timezones

The following time zones are valid to be specified in the `TimeZone` field of the User Resource Item (see section User Resource Item).

Afghanistan Standard Time
Alaskan Standard Time
Arab Standard Time
Arabian Standard Time
Arabic Standard Time
Argentina Standard Time
Atlantic Standard Time
AUS Central Standard Time
AUS Eastern Standard Time
Azerbaijan Standard Time
Azores Standard Time
Canada Central Standard Time
Cape Verde Standard Time
Caucasus Standard Time
Cen. Australia Standard Time
Central America Standard Time
Central Asia Standard Time
Central Brazilian Standard Time
Central Europe Standard Time
Central European Standard Time
Central Pacific Standard Time
Central Standard Time
Central Standard Time (Mexico)
China Standard Time
Dateline Standard Time
E. Africa Standard Time
E. Australia Standard Time
E. Europe Standard Time
E. South America Standard Time
Eastern Standard Time
Egypt Standard Time
Ekaterinburg Standard Time
Fiji Standard Time
FLE Standard Time
Georgian Standard Time
GMT Standard Time
Greenland Standard Time
Greenwich Standard Time
GTB Standard Time
Hawaiian Standard Time
India Standard Time
Iran Standard Time
Jerusalem Standard Time
Jordan Standard Time
Korea Standard Time
Malay Peninsula Standard Time
Mid-Atlantic Standard Time
Middle East Standard Time
Montevideo Standard Time
Morocco Standard Time
Mountain Standard Time
Mountain Standard Time (Mexico)
Myanmar Standard Time
N. Central Asia Standard Time
Namibia Standard Time
Nepal Standard Time
New Zealand Standard Time
Newfoundland Standard Time
North Asia East Standard Time
North Asia Standard Time
Pacific SA Standard Time
Pacific Standard Time
Pacific Standard Time (Mexico)
Pakistan Standard Time
Romance Standard Time
Russian Standard Time
SA Eastern Standard Time
SA Pacific Standard Time
SA Western Standard Time
Samoa Standard Time
SE Asia Standard Time
South Africa Standard Time
Sri Lanka Standard Time
Taipei Standard Time
Tasmania Standard Time
Tokyo Standard Time
Tonga Standard Time
US Eastern Standard Time
US Mountain Standard Time
Venezuela Standard Time
Vladivostok Standard Time
W. Australia Standard Time
W. Central Africa Standard Time
W. Europe Standard Time
West Asia Standard Time
West Pacific Standard Time
Yakutsk Standard Time
Appendix D. Example Notification Endpoint WSDL

This Appendix contains some example WSDL for a subscriber client that expects to receive SOAP requests containing notifications about state changes that the client has previously subscribed to receive.

This is the WSDL implemented by an Unified CCMP test notification endpoint.

```xml
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions name="SinkStatus" targetNamespace="http://www.exony.com/schemas/2009/10/status"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:wsa10="http://www.w3.org/2005/08/addressing"
    xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
    xmlns:ws="http://schemas.xmlsoap.org/wsdl/
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">
  <wsp:Policy wsu:Id="TrustedWebClientBinding_IStatus_policy">
    <ws:ExactlyOne>
      <ws:All>
        <sp:TransportBinding
            xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
          <wsp:Policy>
            <sp:TransportToken>
              <wsp:Policy>
                <sp:HttpsToken RequireClientCertificate="false" />
              </wsp:Policy>
            </sp:TransportToken>
            <sp:AlgorithmSuite>
              <wsp:Policy>
                <sp:Basic256 />
              </wsp:Policy>
            </sp:AlgorithmSuite>
            <sp:Layout>
            </sp:Layout>
          </wsp:Policy>
        </sp:TransportBinding>
      </ws:All>
    </ws:ExactlyOne>
  </wsp:Policy>
</wsdl:definitions>
```
<wsp:Policy>
  <sp:Strict />
</wsp:Policy>
</sp:Layout>
<sp:IncludeTimestamp />
</wsp:Policy>
</sp:TransportBinding>
- <sp:SignedSupportingTokens
  xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
  - <wsp:Policy>
    - <sp:UsernameToken
      sp:IncludeToken="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy/IncludeToken/AlwaysToRecipient">
      - <wsp:Policy>
        <sp:WssUsernameToken10 />
      </wsp:Policy>
    </sp:UsernameToken>
  </wsp:Policy>
  - <sp:EndorsingSupportingTokens
    xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
    - <wsp:Policy>
      - <sp:X509Token
        sp:IncludeToken="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy/IncludeToken/AlwaysToRecipient">
        - <wsp:Policy>
          <sp:RequireThumbprintReference />
          <sp:WssX509V3Token10 />
        </wsp:Policy>
      </sp:X509Token>
      - <sp:SignedParts>
        <sp:Header Name="To"
          Namespace="http://www.w3.org/2005/08/addressing" />
      </sp:SignedParts>
    </wsp:Policy>
    - <sp:Wss11
      xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
      - <wsp:Policy>
        <sp:MustSupportRefKeyIdentifier />
        <sp:MustSupportRefIssuerSerial />
        <sp:MustSupportRefThumbprint />
        <sp:MustSupportRefEncryptedKey />
      </wsp:Policy>
    </sp:Wss11>
  </sp:SignedSupportingTokens>
- <sp:Trust10
  xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
  - <wsp:Policy>
    <sp:MustSupportIssuedTokens />
  </wsp:Policy>
<sp:RequireClientEntropy />
<sp:RequireServerEntropy />
</wsp:Policy>
</sp:Trust10>
</wsaw:UsingAddressing />
</wsp:ExactlyOne>
</wsp:Policy>

-wsdl:types
-<xsd:schema elementFormDefault="qualified"
targetNamespace="http://www.exony.com/schemas/2009/10/status"
-<xsd:element name="OnStatusChange">
-<xsd:complexType>
-<xsd:sequence>
<xsd:element minOccurs="0" name="status" nillable="true" type="tns:ResourceStatus" />
</xsd:sequence>
</xsd:complexType>
</xsd:element>

-xsd:complexType name="ResourceStatus">
-<xsd:sequence>
<xsd:element minOccurs="0" name="ContainerId" type="ser:guid" />
<xsd:element name="Id" nillable="true" type="xsd:string" />
<xsd:element name="Status" nillable="true" type="xsd:string" />
<xsd:element name="Type" nillable="true" type="xsd:string" />
</xsd:sequence>
</xsd:complexType>
</xsd:element>

<xsd:element name="ResourceStatus" nillable="true" type="tns:ResourceStatus" />
</xsd:schema>

-xsd:schema attributeFormDefault="qualified" elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/2003/10/Serialization/
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://schemas.microsoft.com/2003/10/Serialization/">
<xsd:element name="anyType" nillable="true" type="xs:anyType" />
<xsd:element name="anyURI" nillable="true" type="xs:anyURI" />
<xsd:element name="base64Binary" nillable="true" type="xs:base64Binary" />
<xsd:element name="boolean" nillable="true" type="xs:boolean" />
<xsd:element name="byte" nillable="true" type="xs:byte" />
<xsd:element name="dateTime" nillable="true" type="xs:dateTime" />
<xsd:element name="decimal" nillable="true" type="xs:decimal" />
<xsd:element name="double" nillable="true" type="xs:double" />
<xsd:element name="float" nillable="true" type="xs:float" />
<xsd:element name="int" nillable="true" type="xs:int" />
<xsd:element name="long" nillable="true" type="xs:long" />
<xsd:element name="QName" nillable="true" type="xs:QName" />
<xs:element name="short" nillable="true" type="xs:short" />
<xs:element name="string" nillable="true" type="xs:string" />
<xs:element name="unsignedByte" nillable="true" type="xs:unsignedByte" />
<xs:element name="unsignedInt" nillable="true" type="xs:unsignedInt" />
<xs:element name="unsignedLong" nillable="true" type="xs:unsignedLong" />
<xs:element name="unsignedShort" nillable="true" type="xs:unsignedShort" />
<xs:element name="char" nillable="true" type="tns:char" />
- <xs:simpleType name="char">
  <xs:restriction base="xs:int" />
</xs:simpleType>
<xs:element name="duration" nillable="true" type="tns:duration" />
- <xs:simpleType name="duration">
  <xs:restriction base="xs:duration">
    <xs:pattern value="\-?P\(\d*D)?\(\(\d{H}\)?\(\d{M}\)?\(\(\d{S}\)?\)S)\?" />
    <xs:minInclusive value="\-P10675199DT2H48M5.4775808S" />
    <xs:maxInclusive value="P10675199DT2H48M5.4775807S" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="guid" nillable="true" type="tns:guid" />
- <xs:simpleType name="guid">
  <xs:restriction base="xs:string">
  </xs:restriction>
</xs:simpleType>
<xs:attribute name="FactoryType" type="xs:QName" />
<xs:attribute name="Id" type="xs:ID" />
<xs:attribute name="Ref" type="xs:IDREF" />
</xs:schema>
</wsdl:types>
- <wsdl:message name="IStatus_OnStatusChange_InputMessage">
  <wsdl:part name="parameters" element="tns:OnStatusChange" />
</wsdl:message>
- <wsdl:portType name="IStatus">
  - <wsdl:operation name="OnStatusChange">
    <wsdl:input
      message="tns:IStatus_OnStatusChange_InputMessage" />
  </wsdl:operation>
  </wsdl:portType>
- <wsdl:binding name="TrustedWebClientBinding_IStatus" type="tns:IStatus">
  <wsp:PolicyReference URI="#TrustedWebClientBinding_IStatus_policy" />
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
</wsdl:binding>
- `<wsdl:operation name="OnStatusChange">
  <soap12:operation
  - `<wsdl:input>
    `<soap12:body use="literal" />
  </wsdl:input>
</wsdl:operation>
</wsdl:binding>
- `<wsdl:service name="SinkStatus">
  - `<wsdl:port name="TrustedWebClientBinding_IStatus" binding="tns:TrustedWebClientBinding_IStatus">
    `<soap12:address location="https://localhost:33333/Sink" />
  </wsdl:port>
</wsdl:service>`