



# Network Server Provisioning

## Interface Specification

Release 21.0  
Document Version 1

9737 Washingtonian Boulevard, Suite 350  
Gaithersburg, MD 20878  
Tel +1 301.977.9440

[WWW.BROADSOFT.COM](http://WWW.BROADSOFT.COM)

## **BroadWorks® Guide**

### **Copyright Notice**

Copyright© 2014 BroadSoft, Inc.

All rights reserved.

Any technical documentation that is made available by BroadSoft, Inc. is proprietary and confidential and is considered the copyrighted work of BroadSoft, Inc.

This publication is for distribution under BroadSoft non-disclosure agreement only. No part of this publication may be duplicated without the express written permission of BroadSoft, Inc., 9737 Washingtonian Boulevard, Suite 350, Gaithersburg, MD 20878.

BroadSoft reserves the right to make changes without prior notice.

### **Trademarks**

Any product names mentioned in this document may be trademarks or registered trademarks of BroadSoft or their respective companies and are hereby acknowledged.

This document is printed in the United States of America.

## Document Revision History

Release	Version	Reason for Change	Date	Author
14.0	1	Updated document for rebranding.	February 7, 2006	Roberta Boyle
14.0	1	Updated document for Release 14.	June 22, 2006	Michel Tougas
14.0	1	Edited document.	August 4, 2006	Patricia Renaud
14.0	1	Edited changes made to document.	September 1, 2006	Patricia Renaud
14.0	2	Updated section <a href="#">4.3 Using Authentication, Login, and Logout</a> to indicate the login requirements for clients proxied via OCS. Updated section <a href="#">5 OSS Commands for System Provisioning</a> to include the <i>loginId</i> parameter for proxied messages. Added <a href="#">Appendix B: Network Server Errors</a> for EV 46671.	May 15, 2007	Robb Surridge
14.0	2	Edited document.	May 15, 2007	Patricia Renaud
14.0	3	Updated document for version 12 and 13 of the API.	July 8, 2008	Yves Racine
14.0	3	Edited and published document.	July 8, 2008	Patricia Renaud
15.0	1	Updated document for Release 15.0.	July 9, 2008	Simon Cadieux
15.0	1	Edited and published document.	July 14, 2008	Patricia Renaud
15.0	2	Added information about commands: addLinePort, deleteLinePort, modifyLinePort, getLinePorts for EV 66697.	December 8, 2008	Goska Auerbach
15.0	2	Edited changes and published document.	January 15, 2009	Andrea Fitzwilliam
16.0	1	Updated document for Release 16.0.	April 3, 2009	Yves Racine
16.0	1	Edited changes and published document.	May 27, 2009	Andrea Fitzwilliam
17.0	1	Updated document for Release 17.0.	March 12, 2010	Juliette D'Almeida
17.0	1	Edited changes and published document.	March 31, 2010	Andrea Fitzwilliam
17.0	2	Updated section <a href="#">2 Overview</a> for EV 128394.	March 8, 2011	Goska Auerbach
17.0	2	Edited changes and published document.	June 20, 2011	Jessica Boyle
18.0	1	Updated document for Release 18.0.	November 10, 2011	Jean-Francois Laporte
18.0	1	Edited changes and published document.	November 24, 2011	Patricia Renaud
18.0	2	Updated section <a href="#">5 OSS Commands for System Provisioning</a> for EV 145326.	February 8, 2012	Jean-Francois Laporte
18.0	2	Edited changes and published document.	February 20, 2012	Andrea Fitzwilliam
18.0	3	Updated section <a href="#">1.14 Supported Versions</a> .	February 28, 2012	Jean-Francois Laporte
18.0	3	Edited changes and published document.	March 13, 2012	Andrea Fitzwilliam
19.0	1	Updated document for Release 19.0.	October 3, 2012	Jean-Francois Laporte

Release	Version	Reason for Change	Date	Author
19.0	1	Updated section <a href="#">5 OSS Commands for System Provisioning</a> for EV 154458.	November 1, 2012	Jean-Francois Laporte
20.0	1	Updated document for Release 20.0	December 3, 2012	Jean-Francois Laporte
20.0	1	Updated section <a href="#">5 OSS Commands for System Provisioning</a> for EV 179232.	January 10, 2013	Husam Al Sahli
20.0	1	Updated section <a href="#">5 OSS Commands for System Provisioning</a> for EV 151065.	May 12, 2013	Husam Al Sahli
20.0	1	Updated section <a href="#">5 OSS Commands for System Provisioning</a> for EV 189453.	May 29, 2013	Pierre Drapeau
20.0	1	Updated section <a href="#">5 OSS Commands for System Provisioning</a> for EV 193362.	July 9, 2013	Husam Al Sahli
20.0	1	Updated section <a href="#">5 OSS Commands for System Provisioning</a> for EV 196487.	July 11, 2013	Husam Al Sahli
20.0	1	Updated section <a href="#">3 Network Server OCI-P XML Definition</a> for EV 196569.	July 31, 2013	Jean-Francois Laporte
20.0	1	Updated sections <a href="#">1.13 Changes for Previous Releases</a> , <a href="#">1.14 Supported Versions</a> , and <a href="#">5 OSS Commands for System Provisioning</a> for EV 195752.	August 1, 2013	Jean-Francois Laporte
20.0	1	Edited changes and published document.	October 22, 2013	Joan Renaud
21.0	1	Updated document for Release 21.0.	June 6, 2014	Jean-Francois Laporte
21.0	1	Edited changes and updated copyright notice.	October 14, 2014	Joan Renaud
21.0	1	Rebranded and published document.	December 9, 2014	Joan Renaud

## Table of Contents

---

<b>1</b>	<b>Summary of Changes .....</b>	<b>19</b>
1.1	Changes for Release 21.0, Document Version 1 .....	19
1.2	Changes for Release 20.0, Document Version 1 .....	19
1.3	Changes for Release 19.0, Document Version 1 .....	19
1.4	Changes for Release 18.0, Document Version 3 .....	19
1.5	Changes for Release 18.0, Document Version 2 .....	19
1.6	Changes for Release 18.0, Document Version 1 .....	19
1.7	Changes for Release 17.0, Document Version 2 .....	20
1.8	Changes for Release 17.0, Document Version 1 .....	20
1.9	Changes for Release 16.0, Document Version 1 .....	20
1.10	Changes for Release 15.0, Document Version 2 .....	20
1.11	Changes for Release 15.0, Document Version 1 .....	20
1.12	Changes for Releases 14.sp2 and 14.sp3, Document Version 3.....	20
1.13	Changes for Release 14.0, Document Version 2 .....	21
1.14	Changes for Previous Releases .....	21
1.15	Supported Versions .....	23
<b>2</b>	<b>Overview.....</b>	<b>24</b>
2.1	OSS Interface .....	24
2.2	OCI-P Interface.....	24
2.2.1	Authorization Levels and Access Control.....	25
2.2.2	Aggregate OCI Commands .....	25
2.2.3	Validation.....	26
<b>3</b>	<b>Network Server OCI-P XML Definition .....</b>	<b>27</b>
<b>4</b>	<b>Network Server OSS XML Definition.....</b>	<b>29</b>
4.1	Document Type Definition (DTD) .....	29
4.1.1	Element Description .....	42
4.2	Narrow Results of Get Command .....	51
4.2.1	Sample Input XML Structure.....	52
4.2.2	Sample Output XML Structure.....	52
4.3	Using Authentication, Login, and Logout .....	53
4.3.1	requestAuthentication.....	55
4.3.2	requestLogin .....	55
4.3.3	requestLogout.....	56
<b>5</b>	<b>OSS Commands for System Provisioning.....</b>	<b>58</b>
5.1	getAllServiceProviderProfiles.....	58
5.1.1	Description .....	58
5.1.2	Sample Input XML Structure.....	58
5.1.3	Sample Output XML Structure.....	58
5.2	getAllCarriers .....	59

5.2.1	Description .....	59
5.2.2	Sample Input XML Structure.....	59
5.2.3	Sample Output XML Structure.....	59
5.3	getLocalCallingAreas .....	60
5.3.1	Description .....	60
5.3.2	Sample Input XML Structure.....	60
5.3.3	Sample Output XML Structure.....	60
5.4	getAllPrivatePolicies .....	61
5.4.1	Description .....	61
5.4.2	Sample Input XML Structure.....	61
5.4.3	Sample Output XML Structure.....	61
5.5	addPrivatePolicyInstance.....	62
5.5.1	Description .....	62
5.5.2	Sample Input XML Structure.....	62
5.5.3	Sample Output XML Structure.....	63
5.6	deletePrivatePolicyInstance.....	63
5.6.1	Description .....	63
5.6.2	Sample Input XML Structure.....	64
5.6.3	Sample Output XML Structure.....	64
5.7	modifyPrivatePolicyInstance .....	65
5.7.1	Description .....	65
5.7.2	Sample Input XML Structure.....	65
5.7.3	Sample Output XML Structure.....	66
5.8	getPrivatePolicyInstance.....	66
5.8.1	Description .....	66
5.8.2	Sample Input XML Structure.....	67
5.8.3	Sample Output XML Structure.....	67
5.9	synchronizeEnterprise.....	68
5.9.1	Description .....	68
5.9.2	Sample Input XML Structure.....	68
5.9.3	Sample Output XML Structure.....	70
5.10	modifyEnterprise.....	70
5.10.1	Description .....	70
5.10.2	Sample Input XML Structure.....	71
5.10.3	Sample Output XML Structure.....	71
5.11	deleteEnterprise.....	72
5.11.1	Description .....	72
5.11.2	Sample Input XML Structure.....	72
5.11.3	Sample Output XML Structure.....	72
5.12	getEnterprise.....	73
5.12.1	Description .....	73
5.12.2	Sample Input XML Structure.....	73
5.12.3	Sample Output XML Structure.....	73

5.13	getEnterprises.....	74
5.13.1	Description .....	74
5.13.2	Sample Input XML Structure.....	74
5.13.3	Sample Output XML Structure.....	74
5.14	getHostsForEnterprise .....	75
5.14.1	Description .....	75
5.14.2	Sample Input XML Structure.....	76
5.14.3	Sample Output XML Structure.....	76
5.15	addSite .....	76
5.15.1	Description .....	76
5.15.2	Sample Input XML Structure.....	77
5.15.3	Sample Output XML Structure.....	77
5.16	modifySite .....	78
5.16.1	Description .....	78
5.16.2	Sample Input XML Structure.....	78
5.16.3	Sample Output XML Structure.....	78
5.17	deleteSite .....	79
5.17.1	Description .....	79
5.17.2	Sample Input XML Structure.....	79
5.17.3	Sample Output XML Structure.....	79
5.18	getSite .....	80
5.18.1	Description .....	80
5.18.2	Sample Input XML Structure.....	80
5.18.3	Sample Output XML Structure.....	80
5.19	getSites .....	81
5.19.1	Description .....	81
5.19.2	Sample Input XML Structure.....	81
5.19.3	Sample Output XML Structure.....	81
5.20	addHostingNe.....	82
5.20.1	Description .....	82
5.20.2	Sample Input XML Structure.....	83
5.20.3	Sample Output XML Structure.....	84
5.21	deleteHostingNe.....	84
5.21.1	Description .....	84
5.21.2	Sample Input XML Structure.....	84
5.21.3	Sample Output XML Structure.....	85
5.22	modifyHostingNe.....	85
5.22.1	Description .....	85
5.22.2	Sample Input XML Structure.....	86
5.22.3	Sample Output XML Structure.....	87
5.23	getHostingNes .....	87
5.23.1	Description .....	87
5.23.2	Sample Input XML Structure.....	87

5.23.3	Sample Output XML Structure.....	88
5.24	getHostingNe .....	89
5.24.1	Description .....	89
5.24.2	Sample Input XML Structure.....	89
5.24.3	Sample Output XML Structure.....	89
5.25	addHostingNeNode.....	90
5.25.1	Description .....	90
5.25.2	Sample Input XML Structure.....	90
5.25.3	Sample Output XML Structure.....	91
5.26	deleteHostingNeNode .....	91
5.26.1	Description .....	91
5.26.2	Sample Input XML Structure.....	91
5.26.3	Sample Output XML Structure.....	92
5.27	modifyHostingNeNode .....	92
5.27.1	Description .....	92
5.27.2	Sample Input XML Structure.....	92
5.27.3	Sample Output XML Structure.....	93
5.28	getHostingNeNodes .....	93
5.28.1	Description .....	93
5.28.2	Sample Input XML Structure.....	93
5.28.3	Sample Output XML Structure.....	94
5.29	getHostingNeNode .....	95
5.29.1	Description .....	95
5.29.2	Sample Input XML Structure.....	95
5.29.3	Sample Output XML Structure.....	95
5.30	addHostingNeAddr .....	96
5.30.1	Description .....	96
5.30.2	Sample Input XML Structure.....	96
5.30.3	Sample Output XML Structure.....	97
5.31	deleteHostingNeAddr .....	97
5.31.1	Description .....	97
5.31.2	Sample Input XML Structure.....	98
5.31.3	Sample Output XML Structure.....	98
5.32	modifyHostingNeAddr .....	98
5.32.1	Description .....	98
5.32.2	Sample Input XML Structure.....	99
5.32.3	Sample Output XML Structure.....	99
5.33	getHostingNeAddrs .....	100
5.33.1	Description .....	100
5.33.2	Sample Input XML Structure.....	100
5.33.3	Sample Output XML Structure.....	100
5.34	addHostingNEAdministrators.....	101
5.34.1	Description .....	101



5.34.2	Sample Input XML Structure.....	101
5.34.3	Sample Output XML Structure.....	101
5.35	deleteHostingNEAdministrators.....	102
5.35.1	Description .....	102
5.35.2	Sample Input XML Structure.....	102
5.35.3	Sample Output XML Structure.....	102
5.36	getHostingNEAdministrators.....	103
5.36.1	Description .....	103
5.36.2	Sample Input XML Structure.....	103
5.36.3	Sample Output XML Structure.....	103
5.37	addEnterpriseAdmin.....	104
5.37.1	Description .....	104
5.37.2	Sample Input XML Structure.....	104
5.37.3	Sample Output XML Structure.....	104
5.38	deleteEnterpriseAdmin.....	105
5.38.1	Description .....	105
5.38.2	Sample Input XML Structure.....	105
5.38.3	Sample Output XML Structure.....	105
5.39	modifyEnterpriseAdmin.....	106
5.39.1	Description .....	106
5.39.2	Sample Input XML Structure.....	106
5.39.3	Sample Output XML Structure.....	106
5.40	getEnterpriseAdmins.....	107
5.40.1	Description .....	107
5.40.2	Sample Input XML Structure.....	107
5.40.3	Sample Output XML Structure.....	107
5.41	getEnterpriseAdmin.....	108
5.41.1	Description .....	108
5.41.2	Sample Input XML Structure.....	108
5.41.3	Sample Output XML Structure.....	109
5.42	synchronizeUserGroup.....	109
5.42.1	Description .....	109
5.42.2	Sample Input XML Structure.....	110
5.42.3	Sample Output XML Structure.....	110
5.43	assignAddresses.....	111
5.43.1	Description .....	111
5.43.2	Sample Input XML Structure.....	111
5.43.3	Sample Output XML Structure.....	112
5.44	modifyAddresses.....	113
5.44.1	Description .....	113
5.44.2	Sample Input XML Structure.....	113
5.44.3	Sample Output XML Structure.....	114
5.45	unassignAddresses.....	114

5.45.1	Description .....	114
5.45.2	Sample Input XML Structure.....	115
5.45.3	Sample Output XML Structure.....	115
5.46	modifyUserGroup .....	116
5.46.1	Description .....	116
5.46.2	Sample Input XML Structure.....	116
5.46.3	Sample Output XML Structure.....	117
5.47	deleteUserGroup .....	117
5.47.1	Description .....	117
5.47.2	Sample Input XML Structure.....	117
5.47.3	Sample Output XML Structure.....	118
5.48	getUserGroup .....	118
5.48.1	Description .....	118
5.48.2	Sample Input XML Structure.....	118
5.48.3	Sample Output XML Structure.....	118
5.49	getUserGroups .....	120
5.49.1	Description .....	120
5.49.2	Sample Input XML Structure.....	120
5.49.3	Sample Output XML Structure.....	120
5.50	addAuthorizedHost.....	123
5.50.1	Description .....	123
5.50.2	Sample Input XML Structure.....	123
5.50.3	Sample Output XML Structure.....	123
5.51	deleteAuthorizedHost.....	124
5.51.1	Description .....	124
5.51.2	Sample Input XML Structure.....	124
5.51.3	Sample Output XML Structure.....	124
5.52	getAuthorizedHosts .....	125
5.52.1	Description .....	125
5.52.2	Sample Input XML Structure.....	125
5.52.3	Sample Output XML Structure.....	125
5.53	getCallTypes .....	126
5.53.1	Description .....	126
5.53.2	Sample Input XML Structure.....	126
5.53.3	Sample Output XML Structure.....	126
5.54	getTreatments.....	126
5.54.1	Description .....	126
5.54.2	Sample Input XML Structure.....	126
5.54.3	Sample Output XML Structure.....	127
5.55	addPrivatePolicyInstanceDataEntry .....	127
5.55.1	Description .....	127
5.55.2	Sample Input XML Structure.....	128
5.55.3	Sample Output XML Structure.....	129

5.56	addPrivatePolicyPropertiesDataEntry .....	130
5.56.1	Description .....	130
5.56.2	Sample Input XML Structure.....	130
5.56.3	Sample Output XML Structure.....	131
5.57	modifyPrivatePolicyInstanceDataEntry .....	131
5.57.1	Description .....	131
5.57.2	Sample Input XML Structure.....	132
5.57.3	Sample Output XML Structure.....	133
5.58	modifyPrivatePolicyPropertiesDataEntry .....	134
5.58.1	Description .....	134
5.58.2	Sample Input XML Structure.....	134
5.58.3	Sample Output XML Structure.....	135
5.59	getPrivatePolicyInstanceDataEntry .....	135
5.59.1	Description .....	135
5.59.2	Sample Input XML Structure.....	135
5.59.3	Sample Output XML Structure.....	136
5.60	getPrivatePolicyPropertiesDataEntry .....	138
5.60.1	Description .....	138
5.60.2	Sample Input XML Structure.....	138
5.60.3	Sample Output XML Structure.....	138
5.61	deletePrivatePolicyInstanceDataEntry .....	140
5.61.1	Description .....	140
5.61.2	Sample Input XML Structure.....	140
5.61.3	Sample Output XML Structure.....	141
5.62	deletePrivatePolicyPropertiesDataEntry .....	141
5.62.1	Description .....	141
5.62.2	Sample Input XML Structure.....	142
5.62.3	Sample Output XML Structure.....	142
5.63	addUnassignedNumber .....	143
5.63.1	Description .....	143
5.63.2	Sample Input XML Structure.....	143
5.63.3	Sample Output XML Structure.....	143
5.64	deleteUnassignedNumber .....	144
5.64.1	Description .....	144
5.64.2	Sample Input XML Structure.....	144
5.64.3	Sample Output XML Structure.....	145
5.65	getUnassignedNumber .....	145
5.65.1	Description .....	145
5.65.2	Sample Input XML Structure.....	145
5.65.3	Sample Output XML Structure.....	145
5.66	getResourceNes.....	146
5.66.1	Description .....	146
5.66.2	Sample Input XML Structure.....	147

5.66.3	Sample Output XML Structure.....	147
5.67	getResourceNeAdrs.....	148
5.67.1	Description .....	148
5.67.2	Sample Input XML Structure.....	148
5.67.3	Sample Output XML Structure.....	149
5.68	addWebServerFarm.....	149
5.68.1	Description .....	149
5.68.2	Sample Input XML Structure.....	150
5.68.3	Sample Output XML Structure.....	150
5.69	deleteWebServerFarm.....	150
5.69.1	Description .....	150
5.69.2	Sample Input XML Structure.....	150
5.69.3	Sample Output XML Structure.....	151
5.70	modifyWebServerFarm.....	151
5.70.1	Description .....	151
5.70.2	Sample Input XML Structure.....	151
5.70.3	Sample Output XML Structure.....	152
5.71	getWebServerFarms.....	152
5.71.1	Description .....	152
5.71.2	Sample Input XML Structure.....	152
5.71.3	Sample Output XML Structure.....	152
5.72	getWebServerFarm.....	153
5.72.1	Description .....	153
5.72.2	Sample Input XML Structure.....	153
5.72.3	Sample Output XML Structure.....	153
5.73	addWebServerFarmNode.....	154
5.73.1	Description .....	154
5.73.2	Sample Input XML Structure.....	154
5.73.3	Sample Output XML Structure.....	155
5.74	deleteWebServerFarmNode.....	155
5.74.1	Description .....	155
5.74.2	Sample Input XML Structure.....	155
5.74.3	Sample Output XML Structure.....	156
5.75	modifyWebServerFarmNode.....	156
5.75.1	Description .....	156
5.75.2	Sample Input XML Structure.....	156
5.75.3	Sample Output XML Structure.....	157
5.76	getWebServerFarmNodes.....	157
5.76.1	Description .....	157
5.76.2	Sample Input XML Structure.....	157
5.76.3	Sample Output XML Structure.....	157
5.77	getWebServerFarmNode.....	159
5.77.1	Description .....	159

5.77.2	Sample Input XML Structure.....	159
5.77.3	Sample Output XML Structure.....	159
5.78	addWebServerFarmAddr.....	160
5.78.1	Description .....	160
5.78.2	Sample Input XML Structure.....	160
5.78.3	Sample Output XML Structure.....	161
5.79	deleteWebServerFarmAddr.....	161
5.79.1	Description .....	161
5.79.2	Sample Input XML Structure.....	161
5.79.3	Sample Output XML Structure.....	162
5.80	modifyWebServerFarmAddr.....	162
5.80.1	Description .....	162
5.80.2	Sample Input XML Structure.....	162
5.80.3	Sample Output XML Structure.....	163
5.81	getWebServerFarmAddrs.....	163
5.81.1	Description .....	163
5.81.2	Sample Input XML Structure.....	163
5.81.3	Sample Output XML Structure.....	164
5.82	getAllSystemMediaCodec.....	164
5.82.1	Description .....	164
5.82.2	Sample Input XML Structure.....	165
5.82.3	Sample Output XML Structure.....	165
5.83	addNetworkElementMediaCodec.....	166
5.83.1	Description .....	166
5.83.2	Sample Input XML Structure.....	166
5.83.3	Sample Output XML Structure.....	167
5.84	deleteNetworkElementMediaCodec.....	167
5.84.1	Description .....	167
5.84.2	Sample Input XML Structure.....	168
5.84.3	Sample Output XML Structure.....	168
5.85	modifyNetworkElementMediaCodec.....	169
5.85.1	Description .....	169
5.85.2	Sample Input XML Structure.....	169
5.85.3	Sample Output XML Structure.....	170
5.86	getNetworkElementMediaCodec.....	170
5.86.1	Description .....	170
5.86.2	Sample Input XML Structure.....	171
5.86.3	Sample Output XML Structure.....	171
5.87	addDefaultSourceId.....	172
5.87.1	Description .....	172
5.87.2	Sample Input XML.....	172
5.87.3	Sample Output XML.....	172
5.88	modifyDefaultSourceId.....	173

5.88.1	Description .....	173
5.88.2	Sample Input XML .....	173
5.88.3	Sample Output XML .....	173
5.89	deleteDefaultSourceId .....	173
5.89.1	Description .....	173
5.89.2	Sample Input XML .....	173
5.89.3	Sample Output XML .....	174
5.90	getDefaultSourceIds .....	174
5.90.1	Description .....	174
5.90.2	Sample Input XML .....	174
5.90.3	Sample Output XML .....	174
5.91	addLinePort .....	175
5.91.1	Description .....	175
5.91.2	Sample Input XML .....	175
5.91.3	Sample Output XML .....	176
5.92	deleteLinePort .....	176
5.92.1	Description .....	176
5.92.2	Sample Input XML .....	176
5.92.3	Sample Output XML .....	176
5.93	modifyLinePort .....	177
5.93.1	Description .....	177
5.93.2	Sample Input XML .....	177
5.93.3	Sample Output XML .....	178
5.94	getLinePorts .....	178
5.94.1	Description .....	178
5.94.2	Sample Input XML .....	178
5.94.3	Sample Output XML .....	179
5.95	addNetworkTranslationIndex .....	179
5.95.1	Description .....	179
5.95.2	Sample Input XML Structure .....	180
5.95.3	Sample Output XML Structure .....	180
5.96	modifyNetworkTranslationIndex .....	181
5.96.1	Description .....	181
5.96.2	Sample Input XML Structure .....	181
5.96.3	Sample Output XML Structure .....	182
5.97	deleteNetworkTranslationIndex .....	182
5.97.1	Description .....	182
5.97.2	Sample Input XML Structure .....	182
5.97.3	Sample Output XML Structure .....	182
5.98	getNetworkTranslationIndex .....	183
5.98.1	Description .....	183
5.98.2	Sample Input XML Structure .....	183
5.98.3	Sample Output XML Structure .....	183

5.99	getNetworkTranslationIndexes .....	184
5.99.1	Description .....	184
5.99.2	Sample Input XML Structure.....	184
5.99.3	Sample Output XML Structure.....	184
5.100	getPeers.....	185
5.100.1	Description .....	185
5.100.2	Sample Input XML Structure.....	185
5.100.3	Sample Output XML Structure.....	185
5.101	addDataCenter .....	186
5.101.1	Description .....	186
5.101.2	Sample Input XML Structure.....	186
5.101.3	Sample Output XML Structure.....	186
5.102	deleteDataCenter .....	187
5.102.1	Description .....	187
5.102.2	Sample Input XML Structure.....	187
5.102.3	Sample Output XML Structure.....	187
5.103	modifyDataCenter.....	188
5.103.1	Description .....	188
5.103.2	Sample Input XML Structure.....	188
5.103.3	Sample Output XML Structure.....	188
5.104	getDataCenter .....	189
5.104.1	Description .....	189
5.104.2	Sample Input XML Structure.....	189
5.104.3	Sample Output XML Structure.....	189
5.105	getDataCenters.....	190
5.105.1	Description .....	190
5.105.2	Sample Input XML Structure.....	190
5.105.3	Sample Output XML Structure.....	190
5.106	addHostingNESet.....	191
5.106.1	Description .....	191
5.106.2	Sample Input XML Structure.....	191
5.106.3	Sample Output XML Structure.....	191
5.107	deleteHostingNESet.....	192
5.107.1	Description .....	192
5.107.2	Sample Input XML Structure.....	192
5.107.3	Sample Output XML Structure.....	192
5.108	modifyHostingNESet .....	193
5.108.1	Description .....	193
5.108.2	Sample Input XML Structure.....	193
5.108.3	Sample Output XML Structure.....	193
5.109	getHostingNESet .....	194
5.109.1	Description .....	194
5.109.2	Sample Input XML Structure.....	194

5.109.3	Sample Output XML Structure.....	194
5.110	getHostingNESets .....	195
5.110.1	Description .....	195
5.110.2	Sample Input XML Structure.....	195
5.110.3	Sample Output XML Structure.....	195
5.111	addHostingNEHostingNESet.....	196
5.111.1	Description .....	196
5.111.2	Sample Input XML Structure.....	196
5.111.3	Sample Output XML Structure.....	196
5.112	deleteHostingNEHostingNESet.....	197
5.112.1	Description .....	197
5.112.2	Sample Input XML Structure.....	197
5.112.3	Sample Output XML Structure.....	197
5.113	addTrunkGroup .....	198
5.113.1	Description .....	198
5.113.2	Sample Input XML Structure.....	198
5.113.3	Sample Output XML Structure.....	198
5.114	deleteTrunkGroup.....	199
5.114.1	Description .....	199
5.114.2	Sample Input XML Structure.....	199
5.114.3	Sample Output XML Structure.....	199
5.115	modifyTrunkGroup.....	200
5.115.1	Description .....	200
5.115.2	Sample Input XML Structure.....	200
5.115.3	Sample Output XML Structure.....	200
5.116	getTrunkGroup.....	201
5.116.1	Description .....	201
5.116.2	Sample Input XML Structure.....	201
5.116.3	Sample Output XML Structure.....	201
5.117	getTrunkGroups.....	202
5.117.1	Description .....	202
5.117.2	Sample Input XML Structure.....	202
5.117.3	Sample Output XML Structure.....	202
5.118	addNEMaintenancePartition .....	203
5.118.1	Description .....	203
5.118.2	Sample Input XML Structure.....	203
5.118.3	Sample Output XML Structure.....	203
5.119	modifyNEMaintenancePartition .....	204
5.119.1	Description .....	204
5.119.2	Sample Input XML Structure.....	204
5.119.3	Sample Output XML Structure.....	204
5.120	deleteNEMaintenancePartition .....	205
5.120.1	Description .....	205



5.120.2	Sample Input XML Structure.....	205
5.120.3	Sample Output XML Structure.....	205
5.121	getNEMaintenancePartition.....	206
5.121.1	Description.....	206
5.121.2	Sample Input XML Structure.....	206
5.121.3	Sample Output XML Structure.....	206
5.122	getNEMaintenancePartitions.....	207
5.122.1	Description.....	207
5.122.2	Sample Input XML Structure.....	207
5.122.3	Sample Output XML Structure.....	207
5.123	addHostingNENodeNEMaintenancePartition.....	208
5.123.1	Description.....	208
5.123.2	Sample Input XML Structure.....	208
5.123.3	Sample Output XML Structure.....	208
5.124	deleteHostingNENodeNEMaintenancePartition.....	209
5.124.1	Description.....	209
5.124.2	Sample Input XML Structure.....	209
5.124.3	Sample Output XML Structure.....	209
5.125	addWSFNodeNEMaintenancePartition.....	210
5.125.1	Description.....	210
5.125.2	Sample Input XML Structure.....	210
5.125.3	Sample Output XML Structure.....	210
5.126	deleteWSFNodeNEMaintenancePartition.....	211
5.126.1	Description.....	211
5.126.2	Sample Input XML Structure.....	211
5.126.3	Sample Output XML Structure.....	211
5.127	getHostingNesFromGroupId.....	212
5.127.1	Description.....	212
5.127.2	Sample Input XML Structure.....	212
5.127.3	Sample Output XML Structure.....	212
5.128	getEnterpriseTrunkDNRange.....	213
5.128.1	Description.....	213
5.128.2	Sample Input XML Structure.....	213
5.128.3	Sample Output XML Structure.....	213
5.129	getEnterpriseTrunkDNRanges.....	214
5.129.1	Description.....	214
5.129.2	Sample Input XML Structure.....	214
5.129.3	Sample Output XML Structure.....	215
5.130	addEnterpriseTrunkDNRanges.....	215
5.130.1	Description.....	215
5.130.2	Sample Input XML Structure.....	215
5.130.3	Sample Output XML Structure.....	216
5.131	modifyEnterpriseTrunkDNRanges.....	216

5.131.1	Description .....	216
5.131.2	Sample Input XML Structure.....	217
5.131.3	Sample Output XML Structure.....	217
5.132	deleteEnterpriseTrunkDNRanges .....	218
5.132.1	Description .....	218
5.132.2	Sample Input XML Structure.....	218
5.132.3	Sample Output XML Structure.....	218
<b>6</b>	<b>Validation Requirements.....</b>	<b>219</b>
<b>7</b>	<b>Appendix A: Network Server Digit Manipulation .....</b>	<b>224</b>
7.1	Network Server Digit Manipulation Syntax.....	224
7.1.1	Mode of Operation.....	224
7.1.2	Supported Commands .....	224
<b>8</b>	<b>Appendix B: Network Server Errors.....</b>	<b>226</b>
8.1	Network Server Error Codes.....	226
<b>Index</b> .....		<b>237</b>

## 1 Summary of Changes

---

This section describes the changes to this document for each release and document version.

### 1.1 Changes for Release 21.0, Document Version 1

This version of the document includes the following changes:

- Updated section [5 OSS Commands for System Provisioning](#) for EV 198534.
- Updated section [3 Network Server OCI-P XML Definition](#) for EV 198534.

### 1.2 Changes for Release 20.0, Document Version 1

This version of the document includes the following changes:

- Updated section [5 OSS Commands for System Provisioning](#) for EV 179232.
- Updated section [5 OSS Commands for System Provisioning](#) for EV 151065.
- Updated section [5 OSS Commands for System Provisioning](#) for EV 189453.
- Updated section [5 OSS Commands for System Provisioning](#) for EV 193362.
- Updated section [5 OSS Commands for System Provisioning](#) for EV 196487.
- Updated section [3 Network Server OCI-P XML Definition](#) for EV196569.
- Updated sections [1.13 Changes for Previous Releases](#), [1.14 Supported Versions](#), and [5 OSS Commands for System Provisioning](#) for EV195752.

### 1.3 Changes for Release 19.0, Document Version 1

This version of the document includes the following change:

- Added new elements to section [4 Network Server OSS XML Definition](#).

### 1.4 Changes for Release 18.0, Document Version 3

This version of the document includes the following change:

- Updated section [1.14 Supported Versions](#).

### 1.5 Changes for Release 18.0, Document Version 2

This version of the document includes the following change:

- Updated section [5 OSS Commands for System Provisioning](#) for EV 145326.

### 1.6 Changes for Release 18.0, Document Version 1

This version of the document includes the following changes:

- Added new elements to section [4 Network Server OSS XML Definition](#).
- Added *getPeers*, *addDataCenter*, *deleteDataCenter*, *modifyDataCenter*, *getDataCenter*, *getDataCenters*, *addHostingNESet*, *deleteHostingNESet*, *modifyHostingNESet*, *getHostingNESet*, *getHostingNESets*, *addHostingNEHostingNESet*, *deleteHostingNEHostingNESet*, *addTrunkGroup*, *deleteTrunkGroup*, *modifyTrunkGroup*, *getTrunkGroup*, and *getTrunkGroups* to section [5 OSS Commands for System Provisioning](#).

## 1.7 Changes for Release 17.0, Document Version 2

This version of the document includes the following change:

- Updated section [2 Overview](#) for EV 128394.

## 1.8 Changes for Release 17.0, Document Version 1

This version of the document includes the following changes:

- Three new attributes have been added to the *HostingNeData* element. They are used to specify new signaling attributes as follows: *requiresE164Number*, *sessionReplicationEnabled*, and *xspVersionEqual*.
- New *NetworkTranslationIndexData* and related commands were added.

## 1.9 Changes for Release 16.0, Document Version 1

This version of the document includes the following changes:

There is a new attribute in element *HostingNeData* to specify new signaling attribute *requiresChargeIndication*.

## 1.10 Changes for Release 15.0, Document Version 2

This version of the document includes the following changes:

Information about commands *addLinePort*, *deleteLinePort*, *modifyLinePort*, and *getLinePorts* was added for EV 66697.

## 1.11 Changes for Release 15.0, Document Version 1

This version of the document includes the following changes:

Network Server Release 15.0 introduces the Network Server OCI-P provisioning interface to complement the existing OSS interface.

## 1.12 Changes for Releases 14.sp2 and 14.sp3, Document Version 3

### Network Server Release 14.sp3

For the Network Server Release 14.sp3, additional provisioning capabilities have been added to the interface, now at version 13.0. The additions do not impact existing functionality. Additional capabilities include the ability to add, delete, modify, and get default source IDs based on a range of digits.

Version 13.0 of the application programming interface (API) is the only version supported by Network Server Release 14.sp3.

### Network Server Release 14.sp2

For the Network Server Release 14.sp2, additional provisioning capabilities have been added to the interface, now at version 18.0. These additions do not impact existing functionality. Additional capabilities include:

- New attribute in element *HostingNeData* to specify new signaling attributes as follows: *NSReasonSupported*, *isSubLocationRoutingBypass* and *isESLCallCategoryOverride*.
- New *sourceIdForEAPrefix* attribute in element *EnterpriseData*, which supports the incoming and outgoing originating trunk group (OTG) for equal access calls.

Version 18.0 of the API is the only version supported by Network Server Release 14.sp2.

## 1.13 Changes for Release 14.0, Document Version 2

### Network Server Release 14.0

For the Network Server Release 14.0, additional provisioning capabilities have been added to the interface, now at version 10.0. The additions do not impact existing functionality. Additional capabilities include:

- The ability to add, modify, delete, and get *mediaCodecs* assigned to network elements in the system.
- The ability to query all the *mediaCodecs* that are currently available for use in the system.

Version 10.0 of the API is the only version supported by Network Server Release 14.0.

## 1.14 Changes for Previous Releases

### Network Server Release 13.0

For the Network Server Release 13.0, additional provisioning capabilities have been added to the interface, still at version 9.0. The additions do not impact existing functionality. Additional capabilities include:

- Added commands `requestAuthentication`, `requestLogin`, `requestLogout`. Prior to this release, the login occurred as part of retrieving the *OSSSynchronousSession* instance using the `getSession()` method in the *OSSServer* interface. In this release, a new method `getSession2()` is added to retrieve the *OSSSynchronousSession* instance. The authentication, login, and logout are executed as separate XML commands.

Version 9.0 of the API is the only version supported by Network Server Release 13.0.

### Network Server Release 12.0

For the Network Server Release 12.0, additional provisioning capabilities have been added to the interface, now at version 9.0. The additions do not impact existing functionality. Additional capabilities include:

- New transport attribute in element `HostingNeAddrData` and `ResourceNeAddrData` to specify the transport protocol to be used by the hosting NE and the resource NE.
- New commands and elements to manage the Web Server Farm.
- New version, `patchLevel`, and `minimumPatchLevel` attributes in `HostingNeNodeData`.
- New attribute `countrycode` in element `CarrierData` to return the country code associated with a carrier.

Version 9.0 of the API is the only version supported by Network Server Release 12.0.

### Network Server Release 11.1

For the Network Server Release 11.1, additional provisioning capabilities have been added to the interface, now at version 9.0. The additions do not impact existing functionality. Additional capabilities include:

- New commands and elements to manage the hosting NE administrators.
- New `isLoginId` attribute in `UrlData` element.

Version 8.0 of the API is the only version supported by Network Server Release 11.1.

### **Network Server Release 11.0**

For the Network Server Release 11.0, additional provisioning capabilities have been added to the interface, now at version 7.0. The additions do not impact existing functionality. Additional capabilities include:

- New enterprise policy name supported in attribute name of elements PrivatePolicyInstanceData and PrivatePolicyData. The new enterprise policy name is EntMediaSrvSel.
- New sourceId attribute in elements EnterpriseData, SiteData, and UserGroupData for the support of the incoming and outgoing originating trunk group (OTG).
- New Session Initiation Protocol (SIP) signaling option for OTG support on hosting NEs. The new option is found in attribute supportsSourceId of element HostingNeData.
- New Attribute service in element ResourceNeData (for Media Servers).
- Ability to retrieve all local calling area (LCA) IDs in the system.
- New attribute "lcaid" in elements EnterpriseData and UserGroupData to support subscriber-specific LCAs.

Version 7.0 of the Application Programming Interface (API) is the only version supported by Network Server Release 11.0.

### **Network Server Release 10.0**

For the Network Server Release 10.0, additional provisioning capabilities have been added to the interface, now at version 6.0. The additions do not impact existing functionality. Additional capabilities include:

- Complete provisioning of enterprise policies EntNGWRouting and voiceVPN.
- Support of unassigned numbers.
- Ability to retrieve resource Network Elements (NEs) and resource NE addresses.
- New "type" attribute added to hosting NEs.
- New "poll" attribute added to all device types.

### **Network Server Release 9.0**

For the Network Server Release 9.0, a minor addition introduced a change to the OSS interface, bringing its version to 5.0. The addition impacts the provisioning and querying of hosting network elements by introducing a new signaling option.

For the Network Server Release 9.0, address management changed significantly to support redundancy and these enhancements impacted the provisioning interface of hosting network elements.

For the Network Server Release 9.0, the OSS interface, now in version 4.0, includes the following changes:

- The Network Server now supports the concept of site, where a site is used to group users (Directory Numbers [DNs] and Uniform Resource Locators [URLs]) of an enterprise based on their geographic location. The OSS interface supports the creation, deletion, modification, and retrieval of sites.
- The new modifyAddresses command is introduced. This command is used to change the value of attributes assigned to users. Currently, the Network Server supports only one user attribute, namely "site".

- The OSS interface supports a new more explicit schema to represent addresses in the system. The new data elements for DNSs, DN ranges, and URLs are now used in user group elements and they support user attributes (site).
- The *HostingNEData* element has two new attributes for site support.
- The OSS interface supports the provisioning of a new private policy called *EntNGWRouting*. This policy uses the user's site as routing criteria.
- The name and description attributes for the UserGroupData element have been replaced by *groupID* and *groupName* respectively. This change aligns the contents of these fields for the web, CLI, and OSS interfaces.
- The *groupName* attribute can be used to uniquely identify the group for display purposes. If the *groupName* is left blank, it defaults to the format *hostingNE – groupID*.

#### **Network Server Release 8.1**

For the Network Server Release 8.1, the OSS interface (still in version 4.0 because the Document Type Definition [DTD] has not changed), supports the value of “-1” for the hosting NE address port. This value indicates that the port is not returned in the list of contacts.

#### **Network Server Release 6.0**

For the Network Server Release 6.0, the OSS interface provisioning capabilities were greatly expanded to support additional commands and queries. These, added with the previously defined commands and queries, allow a service provider to fully manage enterprises through the OSS interface, thereby almost eliminating the need for the command line interface (CLI).

### **1.15 Supported Versions**

- **Version 9.0** of the API is the only version supported by Network Server Release 13.0.
- **Version 10.0** of the API is the only version supported by Network Server Release 14.0.
- **Version 11.0** of the API is the only version supported by Network Server Release 14.sp1.
- **Version 12.0** of the API is the only version supported by Network Server Release 14.sp2.
- **Version 13.0** of the API is the only version supported by Network Server Releases 14.sp3 and later.
- **Version 14.0** of the API is the only version supported by Network Server Releases 16.0 and later.
- **Version 15.0** of the API is the only version supported by Network Server Releases 17.0 and later.
- **Version 18.0** of the API is the only version supported by Network Server Releases 17.sp1 and later.
- **Version 19.0** of the API is the only version supported by Network Server Releases 19.0 and later.
- **Version 20.0** of the API is the only version supported by Network Server Releases 20.0 and later.

## 2 Overview

---

The Network Server external provisioning functionality is comprised of two separate and complementary interfaces: the Open Client Interface-Provisioning (OCI-P) and the Operations Support System (OSS) interface.

The OSS provisioning interface facilitates the provisioning of enterprises, sites, user groups, and private policies while the OCI-P interface is used to provision various parameters that control routing, public policy instances and profiles, as well as network elements. Both OSS and OCI-P are available on the Provisioning Server (PS), which is part of the Network Server (NS) and the Open Client Server (OCS) that resides on the Xtended Services Platform (Xsp) and the Element Management System (EMS).

### 2.1 OSS Interface

The interface constitutes a standard set of application programming interfaces adapted using an industry standard technology, which allows an OSS to communicate with the Network Server. The interface consists of commands that are used to provision the Network Server system and also to exchange information between the two systems.

The interface has a set of add, modify, and delete commands. In addition, there is a set of query commands to retrieve information from the Network Server to use for a service provider's OSS. The command interface is XML-based.

The Extensible Markup Language (XML) OSS commands between a client application and the Network Server (and the XML OSS responses from the Network Server to the client application) can be transported using the following means:

- **TCP/IP Sockets:** This solution requires the BroadWorks Open Client Server (OCS) to be installed and running on the BroadWorks Element Management System (EMS) (or alternatively, on another type of BroadWorks server). With such a configuration, each client application establishes a Transmission Control Protocol/Internet Protocol (TCP/IP) socket connectivity with the Open Client Server and the Open Client Server then proxies all XML OSS commands to the Network Server that it serves. The Network Server then sends XML OSS responses to the Open Client Server, which dispatches them to the appropriate registered client application. This configuration requires authentication, login, and logout XML commands for the proper management of each client application session. These commands are described in this document.

### 2.2 OCI-P Interface

The Network Server OCI-P interface is an extension to the Network Server OSS interface, in the sense that it is meant to extend the set of data that it is possible to provision with a programmatic interface on the Network Server. The set of configuration information that can be provisioned through the Network Server OCI-P interface includes the following:

- Location
- Carrier
- Call type
- Policy profile
- Dial plan
- NDC
- Zone
- Rate center



- LCA
- DMI
- Routing NE
- Web Server Farm
- Public policies
- Enterprise Trunk Number Ranges

Provisioning messages used by this interface are based on the same OCI schema model as the one used on the Application Server (AS). The OCI is an XML-based application programming interface (API). The interface uses an XML schema to define the structure, content, and semantics of an XML document.

The Network Server OCI is accessible through the following channel:

**OpenClientServer (OCS):** The OpenClientServer can indistinguishably send and receive Network Server OSS and Network Server OCI-P messages over a plain TCP connection to third-party clients. The OCS takes care of the Network Server OSS and Network Server OCI-P registration and of sending or receiving proper messages to the Network Server.

The Network Server OCI-P message structure is based on the base OCI messages. Each Network Server OCI-P must have the protocol field value set to "NSOCI", for example:

```
<xs:attribute name="protocol" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="OCI"/>
      <xs:enumeration value="NSOCI"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
```

### 2.2.1 Authorization Levels and Access Control

Each OCI command has an authorization level and access permission that is used to prevent third-party software from accessing or modifying data that is outside the scope of the login ID. These permissions are taken from the authenticated login ID for the connection established with BroadWorks.

For OCI-P sessions, the authentication level is determined by the credentials granted at the OSS login phase.

All current OCI-P commands require "system" level authentication.

### 2.2.2 Aggregate OCI Commands

Each OCI transaction is an atomic business function that can be performed with no other transactions. In some cases, third-party applications can be set to perform multiple OCI transactions at a single time. For example, when displaying information about a user's profile and retrieving service data, it is possible to create an OCI request with multiple transactions in the body. Note that each transaction is still treated individually and is committed to the datastore individually. It is recommended to limit the number of actions to no more than 15 transactions (to efficiently use server resources).

### **2.2.3 Validation**

The OCI is fully validated at the back end. Only basic validation is performed in the XML schema itself, since BroadWorks is required to provide error messages in multiple languages, which is beyond the capability of the XML schema. All validation performed by the XML schema must be validated by the system sending the transactions, to avoid receiving cryptic XML parsing errors.

## 3 Network Server OCI-P XML Definition

---

The Network Server Open Client Interface-Provisioning (OCI-P) interface is based on a common set of basic messages also used by the Application Server. Note that the use of Rel\_XX is meant to be replaced with the release of interest.

### 3.1 XML Schema Layout

The XML schema is available on Xchange at the following page: *HOME > Support > BroadWorks > BroadWorks Documentation*, then simply select Schema And MIB with the desired release. There you can either access an html version of the schemas or the actual xsds which are contained in *Rel\_XX\_OCISchema\_ALL.zip*. This section describes the structure of the XML schema.

#### 3.1.1 Base XML Schema

The base XML schema is included in the *Rel\_XX\_OCISchemaBASE.zip* file. It contains the base definitions for messages, success and error messages, tables, and other low-level attributes. No functional transactions are defined in these files.

#### 3.1.2 Network Server XML Schema

The Network Server (NS) XML schema is included in the *Rel\_XX\_OCISchemaNS.zip* file. It contains all functional transactions for the system. The following files are available:

- *OCISchemaNS.xsd* – This file is the base file for the Network Server XML schema files. This file references all included files.
- *OCISchemaDataTypes.xsd* – This file contains the data types used in the XML schema that are used at multiple levels in the hierarchy, such as the definitions of an IP address or phone number.
- *OCISchemaDataTypesNS.xsd* – This file contains the data types used in the XML schema that are used at multiple levels in the hierarchy specifically for the Network Server.
- *OCISchemSearchCriteriaNS.xsd* – This file contains the data types used in the XML schema that are included as search criteria to get transactions to filter the returned entries.
- *OCISchemaSystemNS.xsd* – This file contains the transactions that apply for the entire system, such as adding Carriers.
- *OCISchemaRouting.xsd* – This file contains the transactions that apply for parameters that control the routing, like dial plans, zones, rate centers, and so on.
- *OCISchemaRoutingPolicy\*.xsd* – Each policy has a file that contains all transactions that apply for that policy. For example, the *OCISchemaRoutingPolicyCallScreening.xsd* file contains all Call Screening transactions.
- *OCISchemaDeprecated\*.xsd* – These files contains the transactions that have been deprecated from the API over the releases.
- *OCISchemaEnterprise.xsd* – This file contains the data types and transaction specific to enterprises.
- *OCISchemaNetworkElement.xsd* – This file contains the data types and transaction specific to network element.

- *OCISchemaRouting.xsd* – This file contains the data types and transaction specific to system routing on the Network Server.

## 4 Network Server OSS XML Definition

The Network Server Operations Support System (OSS) XML definition provides the schematic description of all OSS commands with attributes that can be used to provision the Network Server. Changes are highlighted in gray.

### 4.1 Document Type Definition (DTD)

```

<?xml version='1.0' encoding="UTF-8"?>
<!ELEMENT com.broadsoft.protocols.nsoss.BroadsoftDocument      (commandArray|command)>
<!ATTLIST com.broadsoft.protocols.nsoss.BroadsoftDocument
clientData  CDATA      #IMPLIED
loginId     CDATA      #IMPLIED
protocol    (NSOSS)   #REQUIRED
version     (21.0)    #REQUIRED
>

<!ELEMENT commandArray      (com.broadsoft.protocols.nsoss.Command+)>

<!ELEMENT command          (result?,commandData?)>
<!ATTLIST command
commandType (requestAuthentication|requestLogin|requestLogout)      #REQUIRED
>
<!ELEMENT commandData      (loginInfo?)>

<!ELEMENT loginInfo        (loginId,reason?,nonce?,algorithm?,password?)>

<!ELEMENT loginId          (#PCDATA)>
<!ELEMENT reason           (#PCDATA)>
<!ELEMENT result           (#PCDATA)>
<!ELEMENT nonce            (#PCDATA)>
<!ELEMENT algorithm        (#PCDATA)>
<!ELEMENT password         (#PCDATA)>

<!ELEMENT com.broadsoft.protocols.nsoss.Command
(commandDataArray?, error?)>
<!ATTLIST com.broadsoft.protocols.nsoss.Command
commandType (addAuthorizedHost|
addDataCenter|
addDefaultSourceId|
addEnterpriseAdmin|
addEnterpriseTrunkDNRRanges|
addHostingNe|
addHostingNEAdministrators|
addHostingNeAddr|
addHostingNEHostingNESet|
addHostingNeNode|
addHostingNENodeNEMaintenancePartition|
addHostingNESet|
addLinePort|
addNEMaintenancePartition|
addNetworkElementMediaCodec|
addNetworkTranslationIndex|
addPrivatePolicyPropertiesDataEntry|
addPrivatePolicyInstance|
addPrivatePolicyInstanceDataEntry|
addSite|
addTrunkGroup|
addWSFNodeNEMaintenancePartition|
addUnassignedNumber|
addWebServerFarm|

```

```
addWebServerFarmAddr|
addWebServerFarmNode|
assignAddresses|
deleteAuthorizedHost|
deleteDataCenter|
deleteDefaultSourceId|
deleteEnterprise|
deleteEnterpriseAdmin|
deleteEnterpriseTrunkDNRanges|
deleteHostingNe|
deleteHostingNEAdministrators|
deleteHostingNeAddr|
deleteHostingNEHostingNESet|
deleteHostingNeNode|
deleteHostingNESet|
deleteLinePort|
deleteNEMaintenancePartition|
deleteHostingNENodeNEMaintenancePartition|
deleteNetworkElementMediaCodec|
deleteNetworkTranslationIndex|
deletePrivatePolicyPropertiesDataEntry|
deletePrivatePolicyInstance|
deletePrivatePolicyInstanceDataEntry|
deleteSite|
deleteTrunkGroup|
deleteUnassignedNumber|
deleteUserGroup|
deleteWebServerFarm|
deleteWebServerFarmAddr|
deleteWebServerFarmNode|
deleteWSFNodeNEMaintenancePartition|
getAllCarriers|
getAllPrivatePolicies|
getAllServiceProviderProfiles|
getAllSystemMediaCodec|
getAuthorizedHosts|
getCallTypes|
getDataCenter|
getDataCenters|
getDefaultSourceIds|
getEnterprise|
getEnterprises|
getEnterpriseAdmin|
getEnterpriseAdmins|
getEnterpriseTrunkDNRange|
getEnterpriseTrunkDNRanges|
getHostingNe|
getHostingNEAdministrators|
getHostingNeAddrs|
getNEMaintenancePartition|
getNEMaintenancePartitions|
getHostingNeNode|
getHostingNeNodes|
getHostingNes|
getHostingNESet|
getHostingNESets|
getHostsForEnterprise|
getLocalCallingAreas|
getLinePorts|
getNetworkElementMediaCodec|
getNetworkTranslationIndex|
getNetworkTranslationIndexes|
getPeers|
```

```

getPrivatePolicyPropertiesDataEntry|
getPrivatePolicyInstance|
getPrivatePolicyInstanceDataEntry|
getResourceNeAdrs|
getResourceNes|
getSite|
getSites|
getTreatments|
getTrunkGroup|
getTrunkGroups|
getUnassignedNumber|
getUserGroup|
getUserGroups|
getWebServerFarm|
getWebServerFarmAdrs|
getWebServerFarmNode|
getWebServerFarmNodes|
getWebServerFarms|
modifyDataCenter|
modifyDefaultSourceId|
modifyEnterprise|
modifyEnterpriseAdmin|
modifyEnterpriseTrunkDNRRanges|
modifyHostingNe|
modifyHostingNeAddr|
modifyHostingNeNode|
modifyHostingNESet|
modifyLinePort|
modifyNEMaintenancePartition|
modifyNetworkElementMediaCodec|
modifyNetworkTranslationIndex|
modifyPrivatePolicyPropertiesDataEntry|
modifyPrivatePolicyInstance|
modifyPrivatePolicyInstanceDataEntry|
modifySite|
modifyTrunkGroup|
modifyUserGroup|
modifyWebServerFarm|
modifyWebServerFarmAddr|
modifyWebServerFarmNode|
synchronizeEnterprise|
synchronizeUserGroup|
unassignAddresses)                                #REQUIRED

elementStartIndex                                CDATA                                #IMPLIED
loginId                                           CDATA                                #IMPLIED
numberOfElements                                CDATA                                #IMPLIED
result                                             CDATA                                #IMPLIED
transactionId                                     CDATA                                #REQUIRED
>

<!ELEMENT commanddataArray
(com.broadsoft.protocols.nsoass.AuthorizedHostData*|
com.broadsoft.protocols.nsoass.CallTypeData*|
com.broadsoft.protocols.nsoass.CarrierData*|
com.broadsoft.protocols.nsoass.DataCenterData*|
com.broadsoft.protocols.nsoass.DefaultSourceIdData*|
com.broadsoft.protocols.nsoass.EnterpriseAdminData*|
com.broadsoft.protocols.nsoass.EnterpriseData*|
com.broadsoft.protocols.nsoass.EnterpriseHostsData*|
com.broadsoft.protocols.nsoass.EnterpriseTrunkingDNRRangeData*|
com.broadsoft.protocols.nsoass.EnterpriseTrunkingDNRangesData*|
com.broadsoft.protocols.nsoass.HostingNeAdminData*|

```

```

com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0*|
com.broadsoft.protocols.nsoss.HostingNeData*|
com.broadsoft.protocols.nsoss.HostingNeNodeData*|
com.broadsoft.protocols.nsoss.HostingNeNodeData_v14_0*|
com.broadsoft.protocols.nsoss.HostingNESetData*|
com.broadsoft.protocols.nsoss.LinePortData*|
com.broadsoft.protocols.nsoss.LCADATA*|
com.broadsoft.protocols.nsoss.NEMaintenancePartitionData*|
com.broadsoft.protocols.nsoss.NetworkElementMediaCodecData*|
com.broadsoft.protocols.nsoss.NetworkTranslationIndexData*|
com.broadsoft.protocols.nsoss.ResourceNeData*|
com.broadsoft.protocols.nsoss.SiteData*|
com.broadsoft.protocols.nsoss.PrivatePolicyData*|
com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData*|
com.broadsoft.protocols.nsoss.ProfileData*|
com.broadsoft.protocols.nsoss.SystemMediaCodecData*|
com.broadsoft.protocols.nsoss.TreatmentData*|
com.broadsoft.protocols.nsoss.TrunkGroupData*|
com.broadsoft.protocols.nsoss.UnassignedNumberData*|
com.broadsoft.protocols.nsoss.UserGroupData*|
com.broadsoft.protocols.nsoss.WebServerFarmData*|
com.broadsoft.protocols.nsoss.WebServerFarmNodeData*|
com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingData*|
com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropertiesData*|
com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesData*|
com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData*|
com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationIns
tanceData*)>

<!-- ** addressArray ***** -->
<!-- ***** -->
<!ELEMENT addressArray
  (com.broadsoft.protocols.nsoss.HostingNeAddrData*|
  com.broadsoft.protocols.nsoss.ResourceNeAddrData*|
  com.broadsoft.protocols.nsoss.WebServerFarmAddrData*)>

<!-- ** AuthorizedHostData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.AuthorizedHostData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.AuthorizedHostData
Policy (EntNGWRouting|FarEndHopOff) #REQUIRED
Enterprise CDATA #REQUIRED
authHost CDATA #IMPLIED
>

<!-- ** CallTypeData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.CallTypeData
  EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.CallTypeData
callType CDATA #REQUIRED
>

<!-- ** CarrierData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.CarrierData
  EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.CarrierData
Name CDATA #REQUIRED
countryCode CDATA #REQUIRED
>

```



```

<!-- ** DataCenterData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.DataCenterData
  EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.DataCenterData
  Name          CDATA          #REQUIRED
  Description   CDATA          #IMPLIED
  >

<!-- ** DefaultSourceIdData***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.DefaultSourceIdData
  EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.DefaultSourceIdData
  fromDn        CDATA          #REQUIRED
  toDn          CDATA          #REQUIRED
  sourceid      CDATA          #IMPLIED
  >

<!-- ** dnArray ***** -->
<!-- ***** -->
<!ELEMENT dnArray          (com.broadsoft.protocols.nsoss.DnData*) >

<!-- ** DnData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.DnData
  EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.DnData
  Dn             CDATA          #REQUIRED
  Site           CDATA          #IMPLIED
  isOffNet       (false|true)  #IMPLIED
  >

<!-- ** dnRangeArray ***** -->
<!-- ***** -->
<!ELEMENT dnRangeArray    (com.broadsoft.protocols.nsoss.DnRangeData*) >

<!-- ** DnRangeData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.DnRangeData
  EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.DnRangeData
  fromDn         CDATA          #REQUIRED
  toDn           CDATA          #REQUIRED
  site           CDATA          #IMPLIED
  isOffNet       (false|true)  #IMPLIED
  >

<!-- ** EnterpriseAdminData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.EnterpriseAdminData
  EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.EnterpriseAdminData
  Enterprise     CDATA          #REQUIRED
  Userid         CDATA          #REQUIRED
  Password       CDATA          #IMPLIED
  firstName      CDATA          #IMPLIED
  lastName       CDATA          #IMPLIED
  >

<!-- ** EnterpriseData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.EnterpriseData

```

```

(privatePoliciesArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.EnterpriseData
Name CDATA #REQUIRED
intraLataCarrier CDATA #IMPLIED
interLataCarrier CDATA #IMPLIED
internationalCarrier CDATA #IMPLIED
description CDATA #IMPLIED
profile CDATA #IMPLIED
callLogging (false|true) #IMPLIED
lcaid CDATA #IMPLIED
sourceId CDATA #IMPLIED
sourceIdForEAPrefix CDATA #IMPLIED
keepGroupsTogether (false|true) #IMPLIED
>

<!-- ** EnterpriseHostsData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.EnterpriseHostsData
(hostsArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.EnterpriseHostsData
enterpriseName CDATA #REQUIRED
>

<!ELEMENT com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData EMPTY>
<!ATTLIST com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData
numberRangeStart CDATA #IMPLIED
numberRangeEnd CDATA #IMPLIED
numberRangeStartPrefix CDATA #IMPLIED
servingAS CDATA #IMPLIED
enterprise CDATA #IMPLIED
site CDATA #IMPLIED
group CDATA #IMPLIED
isOffNet (false|true) #IMPLIED
>

<!ELEMENT com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangesData (dnArray?,
dnRangeArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangesData
servingAS CDATA #REQUIRED
enterprise CDATA #REQUIRED
site CDATA #IMPLIED
group CDATA #IMPLIED
isOffNet (false|true) #IMPLIED
>

<!-- ** EntNGWRoutingPropertiesData ***** -->
<!-- ***** -->
<!ELEMENT
com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropertiesData
EMPTY >
<!ATTLIST
com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropertiesData
callType CDATA #IMPLIED
description CDATA #IMPLIED
>

<!-- ** EntNGWRoutingRoutingData ***** -->
<!-- ***** -->
<!ELEMENT
com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingData
EMPTY >
<!ATTLIST
com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingData

```

```

Enterprise                CDATA                #REQUIRED
Site                      CDATA                #IMPLIED
callType                  CDATA                #IMPLIED
authHost                  CDATA                #IMPLIED
cost                      CDATA                #IMPLIED
weight                    CDATA                #IMPLIED
outDM                     CDATA                #IMPLIED
>

<!-- ** EntSubLocationInstanceData ***** -->
<!-- ***** -->
<!ELEMENT
com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanc
eData
EMPTY>
<!ATTLIST
com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanc
eData
Name                (EntSubLocation)                #REQUIRED
Enterprise          CDATA                #REQUIRED
Enabled             (false|true)                #IMPLIED
intraSiteOnly      (false|true)                #IMPLIED
>

<!-- ** HostingNeAddrData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.HostingNeAddrData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.HostingNeAddrData
Address            CDATA                #REQUIRED
Cost               CDATA                #IMPLIED
Weight            CDATA                #IMPLIED
Port              CDATA                #IMPLIED
Type              (Signaling|Access|DualRouting|Alias) #IMPLIED
Transport         (unspecified|udp|tcp)        #IMPLIED
>

<!-- ** HostingNeAdminData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.HostingNeAdminData
(userIdArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.HostingNeAdminData
hostingNE         CDATA                #REQUIRED
>

<!-- ** HostingNeAdminData_v19_0***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0
(userAdminData?) >
<!ATTLIST com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0
hostingNE         CDATA                #REQUIRED
>

<!-- ** hostingNEArray ***** -->
<!-- ***** -->
<!ELEMENT hostingNEArray (string*) >

<!-- ** HostingNeData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.HostingNeData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.HostingNeData
Name              CDATA                #REQUIRED

```

```

Profile                                CDATA                                #IMPLIED
defaultRoutingEnterprise                CDATA                                #IMPLIED
defaultRoutingSite                    CDATA                                #IMPLIED
defaultSynchEnterprise                 CDATA                                #IMPLIED
defaultSynchSite                      CDATA                                #IMPLIED
supportsE164                          (false|true)                        #IMPLIED
isCallTypeInfoRequired                 (false|true)                        #IMPLIED
supportsPhoneContext                   (false|true)                        #IMPLIED
supportsForking                       (false|true)                        #IMPLIED
supportsSourceId                      (false|true)                        #IMPLIED
countryCode                           CDATA                                #IMPLIED
state                                  CDATA                                #IMPLIED
poll                                   (false|true)                        #IMPLIED
type                                   CDATA                                #IMPLIED
isNSReasonSupported                   (false|true)                        #IMPLIED
isSubLocationRoutingBypass            (false|true)                        #IMPLIED
isESLCallCategoryOverride             (false|true)                        #IMPLIED
requiresChargeIndication              (false|true)                        #IMPLIED
requiresE164Number                    (false|true)                        #IMPLIED
xspVersionEqual                       (false|true)                        #IMPLIED
sessionReplicationEnabled              (false|true)                        #IMPLIED
clusterType                           (primarySecondary|loadBalancing)   #IMPLIED
userCapacityThousands                 CDATA                                #IMPLIED
provisioningCapable                   (false|true)                        #IMPLIED
callProcessingCapable                 (false|true)                        #IMPLIED
sipIpv4NotSupported                   (false|true)                        #IMPLIED
sipIpv6NotSupported                   (false|true)                        #IMPLIED
signalCicAddedAsCacToAs              (false|true)                        #IMPLIED
>

<!-- ** HostingNeNodeData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.HostingNeNodeData
(addressArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.HostingNeNodeData
Name                                CDATA                                #REQUIRED
nodeId                             CDATA                                #REQUIRED
description                         CDATA                                #IMPLIED
isDefault                          (false|true)                        #IMPLIED
version                             CDATA                                #IMPLIED
patchLevel                          CDATA                                #IMPLIED
minimumPatchLevel                  CDATA                                #IMPLIED
loadBalancingWeight                CDATA                                #IMPLIED
dataCenterId                       CDATA                                #IMPLIED
>

<!-- ** HostingNeNodeData_v14_0 ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.HostingNeNodeData_v14_0
(addressArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.HostingNeNodeData_v14_0
Name                                CDATA                                #REQUIRED
nodeId                             CDATA                                #REQUIRED
description                         CDATA                                #IMPLIED
isDefault                          (false|true)                        #IMPLIED
version                             CDATA                                #IMPLIED
webImpactingPatches                CDATA                                #IMPLIED
minimumPatchLevel                  CDATA                                #IMPLIED
>

<!-- ** HostingNESetData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.HostingNESetData

```

```

(hostingNEArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.HostingNESetData
Name                CDATA                #REQUIRED
systemSet           (false|true)        #IMPLIED
>

<!-- ** hostsArray ***** -->
<!-- ***** -->
<!ELEMENT hostsArray (string*) >

<!-- ** LCAData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.LCAData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.LCAData
Lcaid                CDATA                #REQUIRED
>

<!-- ** LinePortData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.LinePortData
EMPTY>
<!ATTLIST com.broadsoft.protocols.nsoss.LinePortData
groupID              CDATA                #IMPLIED
hostingNE            CDATA                #IMPLIED
linePort             CDATA                #IMPLIED
loginIdUrl           CDATA                #IMPLIED
newLinePort          CDATA                #IMPLIED
supportsOrigDiversion (false|true)        #IMPLIED
>

<!-- ** mediaCodecArray ***** -->
<!-- ***** -->
<!ELEMENT mediaCodecArray (com.broadsoft.protocols.nsoss.MediaCodecData*) >

<!-- ** MediaCodecData***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.MediaCodecData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.MediaCodecData
mediaCodecName       CDATA                #REQUIRED
priority             CDATA                #IMPLIED
>

<!-- ** NEMaintenancePartitionData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
(nodeArray?)>
<!ATTLIST com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
Name                CDATA                #REQUIRED
systemPartition     (false|true)        #IMPLIED
>

<!-- ** NetworkElementMediaCodecData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.NetworkElementMediaCodecData
(mediaCodecArray?)>
<!ATTLIST com.broadsoft.protocols.nsoss.NetworkElementMediaCodecData
codecBasedRtgSupported (false|true)        #IMPLIED
neName              CDATA                #REQUIRED
neType              (hosting|routing|resource) #REQUIRED
>

```

```

<!-- ** NetworkTranslationIndexData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
(privatePoliciesArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
Nti                CDATA                #REQUIRED
Enterprise         CDATA                #IMPLIED
Site               CDATA                #IMPLIED
defaultTranslationDN CDATA                #IMPLIED
intraLataCarrier   CDATA                #IMPLIED
interLataCarrier   CDATA                #IMPLIED
internationalCarrier CDATA                #IMPLIED
profile            CDATA                #IMPLIED
lcaid              CDATA                #IMPLIED
>

<!-- ** nodeArray ***** -->
<!-- ***** -->
<!ELEMENT nodeArray (com.broadsoft.protocols.nsoss.NodeData*) >

<!-- ** NodeData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.NodeData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.NodeData
Name                CDATA                #REQUIRED
nodeId              CDATA                #REQUIRED
>

<!-- ** ProfileData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.ProfileData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.ProfileData
Name                CDATA                #REQUIRED
>

<!-- ** AuthorizedHostData ***** -->
<!-- ***** -->
<!ELEMENT privatePoliciesArray (string*) >

<!-- ** PrivatePolicyData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.PrivatePolicyData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.PrivatePolicyData
name                (EntNGWRouting|EntSIMPLE|
                   EntSubLocation|ExtDialing|
                   FarEndHopOff|voiceVPN|
                   EntMediaSrvSel)          #REQUIRED
>

<!-- ** PrivatePolicyInstanceData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData
name                (EntNGWRouting|EntSIMPLE|
                   EntSubLocation|ExtDialing|
                   FarEndHopOff|voiceVPN|
                   EntMediaSrvSel)          #REQUIRED
Enterprise         CDATA                #REQUIRED
Enabled            (false|true)         #IMPLIED

```

```

>
<!-- ** ResourceNeAddrData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.ResourceNeAddrData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.ResourceNeAddrData
Address          CDATA          #REQUIRED
Cost             CDATA          #IMPLIED
Port            CDATA          #IMPLIED
Weight          CDATA          #IMPLIED
Transport        (unspecified|udp|tcp) #IMPLIED
>

<!-- ** ResourceNeData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.ResourceNeData
(addressArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.ResourceNeData
Default          (false|true)      #IMPLIED
defaultCost      CDATA          #IMPLIED
defaultWeight    CDATA          #IMPLIED
location         CDATA          #IMPLIED
name             CDATA          #REQUIRED
poll             (false|true)      #IMPLIED
service          CDATA          #IMPLIED
state           CDATA          #IMPLIED
staticCost       CDATA          #IMPLIED
staticWeight     CDATA          #IMPLIED
type            CDATA          #IMPLIED
>

<!-- ** SiteData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.SiteData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.SiteData
Enterprise       CDATA          #REQUIRED
Name            CDATA          #REQUIRED
Description      CDATA          #IMPLIED
Profile         CDATA          #IMPLIED
sourceId        CDATA          #IMPLIED
>

<!-- ** SystemMediaCodecData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.SystemMediaCodecData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.SystemMediaCodecData
mediaCodec      CDATA          #REQUIRED
>

<!-- ** TreatmentData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.TreatmentData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.TreatmentData
treatmentID     CDATA          #REQUIRED
>

<!-- ** TrunkGroupData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.TrunkGroupData

```

```

EMPTY>
<!ATTLIST com.broadsoft.protocols.nsoss.TrunkGroupData
Name                CDATA                #REQUIRED
groupId             CDATA                #REQUIRED
hostNE              CDATA                #REQUIRED
newName            CDATA                #IMPLIED
identity            CDATA                #IMPLIED
otgDtgIdentity     CDATA                #IMPLIED
>

<!-- ** UserGroupData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.UnassignedNumberData
(dNArray?, dNRangeArray?)>

<!-- ** UserGroupData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.UserGroupData
(dNArray?, dNRangeArray?, uRLArray?)>
<!ATTLIST com.broadsoft.protocols.nsoss.UserGroupData
groupId             CDATA                #REQUIRED
hostingNE          CDATA                #REQUIRED
enterprise         CDATA                #IMPLIED
intraLataCarrier   CDATA                #IMPLIED
interLataCarrier   CDATA                #IMPLIED
internationalCarrier CDATA                #IMPLIED
groupName          CDATA                #IMPLIED
profile            CDATA                #IMPLIED
lcaid              CDATA                #IMPLIED
sourceId           CDATA                #IMPLIED
dNPoolType         CDATA                #IMPLIED
                   (IMRN|SCC)
>

<!-- ** uRLArray ***** -->
<!-- ***** -->
<!ELEMENT uRLArray
(com.broadsoft.protocols.nsoss.UrlData*) >

<!-- ** UrlData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.UrlData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.UrlData
url                CDATA                #REQUIRED
site               CDATA                #IMPLIED
isLoginId          CDATA                #REQUIRED
                   (false|true)
>

<!-- ** VoiceVPNRoutingData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
EMPTY >
<!ATTLIST com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
Enterprise         CDATA                #REQUIRED
locCode            CDATA                #IMPLIED
extMin             CDATA                #IMPLIED
extMax             CDATA                #IMPLIED
selector           CDATA                #IMPLIED
outDM              CDATA                #IMPLIED
hostingNE          CDATA                #IMPLIED
groupId            CDATA                #IMPLIED
treatmentID        CDATA                #IMPLIED
description         CDATA                #IMPLIED
>

```



```

<!-- ** VoiceVPNPropertiesData ***** -->
<!-- ***** -->
<!ELEMENT
com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesData
EMPTY >
<!ATTLIST
com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesData
treatmentID          CDATA          #IMPLIED
description           CDATA          #IMPLIED
>

<!-- ** userAdminArray ***** -->
<!-- ***** -->
<!ELEMENT userAdminArray (com.broadsoft.protocols.nsoss.UserAdminData*) >

<!-- ** UserAdminData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.UserAdminData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.UserAdminData
userId                CDATA          #REQUIRED
groupId               CDATA          #IMPLIED
enterprise             CDATA          #IMPLIED
>

<!-- ** userIdArray ***** -->
<!-- ***** -->
<!ELEMENT userIdArray (string*) >

<!-- ** WebServerFarmAddrData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.WebServerFarmAddrData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.WebServerFarmAddrData
Address                CDATA          #REQUIRED
Cost                   CDATA          #IMPLIED
Weight                 CDATA          #IMPLIED
Type                   CDATA          #IMPLIED
                      (Access|Alias)
>

<!-- ** WebServerFarmData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.WebServerFarmData
EMPTY >
<!ATTLIST com.broadsoft.protocols.nsoss.WebServerFarmData
Name                   CDATA          #REQUIRED
Description            CDATA          #IMPLIED
>

<!-- ** WebServerFarmNodeData ***** -->
<!-- ***** -->
<!ELEMENT com.broadsoft.protocols.nsoss.WebServerFarmNodeData
(addressArray?) >
<!ATTLIST com.broadsoft.protocols.nsoss.WebServerFarmNodeData
Name                   CDATA          #REQUIRED
nodeId                 CDATA          #REQUIRED
state                  CDATA          #IMPLIED
description             CDATA          #IMPLIED
isBrandingMaster       CDATA          #IMPLIED
version                CDATA          #IMPLIED
patchLevel             CDATA          #IMPLIED
minimumPatchLevel      CDATA          #IMPLIED

```

```

>
<!-- ** string ***** -->
<!-- ***** -->
<!ELEMENT string      EMPTY >
<!ATTLIST string
value                  CDATA          #IMPLIED
>

<!-- ** error ***** -->
<!-- ***** -->
<!ELEMENT error      EMPTY >
<!ATTLIST error
Id                    CDATA          #REQUIRED
Summary              CDATA          #REQUIRED
Detail               CDATA          #IMPLIED

```

#### 4.1.1 Element Description

Element	Description
protocol	The XML protocol. This is mandatory and the value must be set to "NSOSS".
version	The version of this DTD. This is mandatory within the DTD.
loginId	Contains the BroadWorks login ID that should be used to execute the Network Server OSS commands. This attribute is required if OSS requests are proxied using the Open Client Server.
clientData	This optional attribute allows a client application to send some data along with an OSS request and to receive it back in the corresponding OSS response. This attribute is useful for correlation purposes when the client application does not wait for a response before sending further OSS requests to the Network Server.
nonce	String returned by the Network Server for password encryption purposes.
password	The client application provides the BroadWorks login ID password using this element.
algorithm	The Network Server uses this attribute to specify the algorithm that the client application shall use to encrypt the BroadWorks login ID password. Only MD5 is currently supported.
reason	When a request for authentication, login, or logout fails, this element contains a textual description of the reason why it failed.
commandArray	Sequence of commands to be executed.
command	Command to be executed and pointing to a unique transaction within the Network Server.
commandType	Command to be executed.
transactionId	A unique numeric identifier within the Network Server that identifies the transaction. This number is used to identify the corresponding response.

Element	Description
elementStartIndex	To narrow down the number of rows that are returned, the client provides two attributes: elementStartIndex and numberOfElements. The elementStartIndex attribute is a numeric value that specifies the first element that will be returned, index 0 being the first element.
numberOfElements	To narrow the number of rows that are returned, the client provides two attributes: elementStartIndex and numberOfElements. The numberOfElements attribute is a numeric value that specifies how many elements to return, including the element at elementStartIndex.
result	The return status of a command. This is an integer value greater than or equal to "0" (zero). A "0" (zero) value implies success. Any other value gives the number of records that failed within the command.
commanddataArray	Nested elements containing 0 or more instances of data that is required for the command.
enterpriseData	An entity that contains enterprise data including the list of provisioned private routing policies.
enterpriseData:name, userGroupData:enterprise, enterpriseAdminData:enterprise, privatePolicyInstanceData:enterprise, authorizedHostData:enterprise, hostingNeData:defaultSynchEnterprise hostingNeData:defaultRoutingEnterprise, siteData:enterprise, EntNGWRoutingRoutingData:enterprise, VoiceVPNRoutingData:enterprise, NetworkTranslationIndexData:enterprise, UserAdminData:enterprise	A unique alphanumeric string identifying an enterprise within the Network Server.
privatePoliciesArray	A list of private routing policies assigned to the enterprise.
intraLataCarrier, NetworkTranslationIndexData: intraLataCarrier	The name of the preferred carrier for intraLATA calls. Set to "NILCAC" if not provided.
interLataCarrier, NetworkTranslationIndexData: interLataCarrier	The name of the preferred carrier for interLATA calls. Set to "NILCAC" if not provided.
internationalCarrier, NetworkTranslationIndexData: internationalCarrier	The name of the preferred carrier for international calls. Set to "NILCAC" if not provided.
description	Description of the enterprise, hosting NE node, Web Server Farm, Web Server Farm node, site, <i>EntNGWRoutingProperties</i> entry, Voice VPN Routing entry, Voice VPN properties or <i>DataCenterData</i> . This attribute is optional.
profile, profileData:name, NetworkTranslationIndexData:profile	A unique alphanumeric string identifying a routing profile within the Network Server.
callLogging	Boolean indication (true/false) assigned to the enterprise to enable or disable logging of calls on an enterprise basis.
userGroupData	A business user group containing directory numbers (individual numbers and/or number ranges) and URLs.

Element	Description
LinePortData:groupID TrunkGroupData:groupID userGroupData:groupID VoiceVPNRoutingData:groupID UserAdminData:groupID	A unique alphanumeric string identifying a user group within a given Application Server or other hosting network element (NE).
userGroupData:groupName	A unique alphanumeric string identifying a user group within the Network Server. This parameter is optional, and if not provided, the default value is "hostingNE – groupID".
siteData	An entity that contains enterprise site data.
siteData:name, hostingNeData:defaultRoutingSite hostingNeData:defaultSynchSite dnData:site dnRangeData:site urlData:site EntNGWRoutingRoutingData:site, NetworkTranslationIndexData:site	A unique alphanumeric string identifying an enterprise site within the Network Server.
hostingNE, hostingNeData:name, hostingNeNodeData:name hostingNeNodeData_v14_0:name authorizedHostData:authHost userGroupData:hostingNE EntNGWRoutingRoutingData:authHost VoiceVPNRoutingData:hostingNE hostingNeAdminData:hostingNE HostingNeAdminData_v19_0:hostingNE LinePortData:hostingNE TrunkGroupData:hostNE	A unique alphanumeric string identifying an Application Server or other hosting NE within the Network Server.
dNArray	A list of unique numbers identifying endpoints.
uRLArray	A list of unique URLs identifying endpoints.
dNRangeArray	A list of paired numbers, representing the lower and upper limits (inclusive) of ranges of directory numbers. The lower and upper limits of a given range must have the same length.
dnData	An entity that contains a unique E.164 directory number along with the enterprise site in which it belongs.
urlData	An entity that contains a unique URL along with the enterprise site in which it belongs.
dnRangeData	An entity that contains a range of sequential E.164 directory numbers along with the enterprise site in which the DNS belong.
dn fromDn toDn	An E.164 directory number used to represent a user in the Network Server.
url	A URL used to represent a user in the Network Server.
isLoginId	Boolean indicating if the user URL is a login ID (true) or an alias (false).
carrierData	An entity that contains carrier data.
carrierData:name	A unique alphanumeric string identifying a carrier within the Network Server.

Element	Description
carrierData:countryCode	The country code of a carrier.
enterpriseAdminData	An entity that contains data related to an enterprise administrator within the Network Server.
userid	A unique alphanumeric string identifying an enterprise administrator within the Network Server.
password	Password for the enterprise administrator.
firstName	The first name of the enterprise administrator. This attribute is optional.
lastName	The last name of enterprise administrator. This attribute is optional.
hostingNeData	An entity that contains data related to an Application Server or other hosting NE within the Network Server.
HostingNeAdminData HostingNeAdminData_v19_0	An entity that contains the information of the administrators of a hosting NE.
userIdArray	A list of hosting NE administrator login IDs.
supportsForking	Boolean indication (true/false) assigned to the hosting NE, to specify if the NE can send parallel INVITE messages upon receiving a list of contacts from the Network Server.
supportsE164	Boolean indication (true/false) assigned to the hosting NE, to specify if the NE can handle directory numbers encoded as described in standard E.164.
isCallTypeInfoRequired	Boolean indication (true/false) assigned to the hosting NE, to specify if the NE is able to handle call type and call category information transmitted by the Network Server. This option should only be enabled for Application Servers. The Application Server uses the call type and call category information to populate call detail recordings.
supportsPhoneContext	Boolean indication (true/false) assigned to the hosting NE, to specify if the NE is able to handle the SIP phone context.
hostingNeData:countryCode	The country code assigned to the hosting NE for the purpose of identifying the origin of calls coming from the NE.
state	Operational state of an NE or a Web Server Farm node. The state can be either <i>OffLine</i> or <i>OnLine</i> . The Network Server excludes OffLine NE nodes from contact lists.
hostingNeNodeData	An entity that contains data related to an Application Server node or other hosting NE node within the Network Server.
addressArray	A list of address information identifying all possible addresses of a hosting NE node, a resource NE node, or a Web Server Farm node.
hostingNeNodeData:nodeId hostingNeNodeData_v14_0:nodeId NodeData:nodeId	In a network configured for redundancy, each replicated hosting NE node is assigned a node ID number. All nodes of the same hosting NE logically define one hosting NE.
isDefault	Boolean indication (true/false) assigned to the hosting NE node, to specify the default initial node for new users provisioned on the hosting NE.

Element	Description
hostingNeNodeData:version hostingNeNodeData_v14_0:version WebServerFarmNodeData:version	Alphanumeric string identifying the version of the hosting NE or Web Server. The format of this field as follows: <type>_<release>_<id> where: “type” is either Application Server or Web Server, “release” is the major release number (for example, Rel_18.0), and “id” is the minor release number (for example, 1.122).
patchLevel	Alphanumeric string identifying the patch level of the hosting NE or Web Server (for example, MP5).
minimumPatchLevel	Alphanumeric string identifying the minimum patch level of the hosting NE, for example, MP5.
minimumPatchLevel	Alphanumeric string identifying the minimum patch level of the Web Server (for example, MP5).
isBrandingMaster	Indicator of whether a Web Server node is a branding master or not. A branding master is the only Web Server capable of changing the branding on an Application Server.
hostingNeAddrData	An entity that contains data related to a hosting NE node address within the Network Server.
address	String identifying one address of a NE node or a Web Server Farm node.
cost	Relative cost of an address on a NE node or a Web Server Farm node. The lower the cost, the higher the preference for using the address over another.
weight	Relative weight of an address on a NE node or a Web Server Farm node. The weight is used to condition the probability of using one address over another. If the weight increases, the frequency of usage of the address also increases.
port	The port associated to a NE node address. If the value is “-1”, the port is not returned in the list of contacts.
poll	Boolean indication (true/false) assigned to a NE, to specify if the NE node should be polled on a regular basis by the Network Server.
type	The type of hosting NE node address, type of hosting NE, type of resource NE, or type of Web Server Farm node address.
transport	Type of transport protocol used to carry SIP signaling supported by the hosting NE node, resource NE node, or the Web Server Farm node. It could be either User Datagram Protocol (UDP), Transmission Control Protocol (TCP), or unspecified.
privatePolicyInstanceData	An entity that contains data describing a private routing policy instance in a Network Server.
privatePolicyInstanceData:name, authorizedHostData:policy, privatePolicyData:name	A unique alphanumeric string identifying a private routing policy within the Network Server.
enabled	Boolean indication (true/false) of a routing policy instance used to indicate if the instance is turned on or off in the Network Server.

Element	Description
authorizedHostData	An entity that contains a hosting NE to be used by a FarEndHopOff or EntNGWRouting policy instance.
profileData	An entity that contains routing profile data.
privatePolicyData	A private routing policy available to be assigned to an enterprise within the Network Server.
EntNGWRoutingRoutingData	An element containing a routing entry for an EntNGWRouting policy instance.
EntNGWRoutingPropertiesData	An element containing a properties entry for the EntNGWRouting policy.
VoiceVPNRoutingData	An element containing a routing entry for a voiceVPN policy instance.
VoiceVPNPropertiesData	An element containing a properties entry for the voiceVPN policy.
TreatmentData	An element containing a treatment ID in the system.
CallTypeData	An element containing a call type ID in the system.
UnassignedNumberData	An element containing phone numbers and/or phone number ranges for the purpose of unassigned number management.
treatmentID	A treatment ID name, as defined in the Network Server. String [1, 8] characters.
callType	A call type ID name, as defined in the Network Server. String [1, 4] characters.
outDM	A digit manipulation string containing digit manipulation instructions (see Appendix). String [0, 80] characters.
locCode	For the voiceVPN policy, it identifies the location in an enterprise dialing plan. String [1, 15] characters.
extMin	For the voiceVPN policy, it identifies the minimum length for an extension. Integer [0, 100].
extMax	For the voiceVPN policy, it identifies the maximum length for an extension. Integer [0, 100].
selector	For the voiceVPN policy, it identifies a routing entry type. Allowed values are "PRIVATE", "PUBLIC", "ROUTE", or "TRMT".
error	Element that is a container for describing problems and issues. It has three attributes: ID, summary, and detail.
id	A numeric identifier for error messages within the Network Server.
summary	Summary of the error.
detail	Detailed description of the error.
default	Boolean indication (true/false) assigned to a resource NE, to specify if the NE is a default Media Server.
defaultCost	Relative cost of a default Media Server. The lower the cost, the higher the preference for using one Media Server over another.

Element	Description
defaultWeight	Relative weight of a default Media Server. The weight is used to condition the probability of using one Media Server over another. If the weight increases, the frequency of usage of the Media Server also increases.
location	String of digits that identifies the location of a resource NE.
ResourceNeData:name	A unique alphanumeric string identifying a resource NE within the Network Server.
staticCost	Relative static cost of a resource NE. Lower the cost, the higher the preference for using one resource NE over another.
staticWeight	Relative static weight of a resource NE. The weight is used to condition the probability of using one resource NE over another. If the weight increases, the frequency of usage of the resource NE also increases.
ResourceNeData	An entity that contains data related to a resource NE within the Network Server.
ResourceNeAddrData	An entity that contains data related to a resource NE address within the Network Server.
LCAData	This element, returned by the Network Server, contains the name of a Local Calling Area (LCA) ID in the system.
Lcaid, NetworkTranslationIndexData:lcaid	This attribute specifies an LCA ID for a Network Server subscriber.
service	Attribute for the ResourceNeData element that lists all Media services that a Media Server supports when used as the default Media Server in the system. If the Media Server supports more than one service, service names are separated with a comma.
sourceId	When a subscriber can be associated with a source ID OTG, then this attribute specifies the source ID name.
sourceIdForEAPrefix	When a subscriber can be associated with a source ID OTG, then this attribute specifies the source ID name. This attribute is only be used for equal access calls.
supportsSourceId	This attribute specifies if a hosting NE supports the <i>OTG</i> parameter received from the Network Server in 302 responses.
isNSReasonSupported	This attribute specifies if a hosting NE sends a reason header with the BW-NS protocol as part of the 302 responses.
isSubLocationRoutingBypass	When a hosting NE is selected by the Subscriber Location policy, if the option is set, the policy simply returns the control to the next policy on the service bus. If the option is not set, a 302 is returned immediately.
isESLCallCategoryOverride	This option is specific to the Enterprise Subscriber Location policy. When this option is set and the Enterprise subscriber location finds a result that is bypassed by the <i>SubLocationRoutingBypass</i> option, the Network Server still marks the call as "private" (call category is set to "private").
webServerFarmData	An entity that contains data related to a Web Server Farm within the Network Server.
webServerFarmData:name	A unique alphanumeric string identifying a Web Server Farm within the Network Server.



Element	Description
webServerFarmNodeData	An entity that contains data related to a Web Server Farm node within the Network Server.
webServerFarmNodeData:name	A unique alphanumeric string identifying a Web Server Farm node within the Network Server.
webServerFarmNodeData:nodeId	In a network configured for redundancy, each replicated Web Server Farm node is assigned a node ID number. All nodes in the same Web Server Farm logically define one Web Server Farm.
webServerFarmAddrData	An entity that contains data related to a Web Server Farm address within the Network Server.
SystemMediaCodecData	Element used to describe a <i>mediaCodec</i> defined in the system.
mediaCodec	Name of a <i>mediaCodec</i> in the system.
NetworkElementMediaCodecData	Element used to describe all <i>mediaCodec</i> -related attributes on a per-network element basis.
NetworkElementMediaCodecData:neName	The name of a network element.
NetworkElementMediaCodecData:neType	The network element type. This is required because the same name can be used for network elements of different type.
codecBasedRtgSupported	A Boolean flag that indicates whether a network element supports codec-based routing. A network element that supports codec-based routing might see its contacts being shuffled in a call processing response, depending on the call's SDP content, and on the <i>mediaCodecs</i> that it supports.
mediaCodecArray	Element that is a container for <i>mediaCodec</i> names.
DefaultSourceIdData	An entity that contains a <i>sourceId</i> associated with a DN range.
HostingNeData:requiresE164Number	Boolean indication (true/false) assigned to the hosting NE, to specify if the NE requires that the directory numbers be encoded as described in standard E.164.
HostingNeData:xspVersionEqual	Boolean indication (true/false) assigned to the hosting NE, to specify whether the Xtended Services Platform version is the same as the Application Server version.
HostingNeData:sessionReplicationEnabled	Boolean indication (true/false) assigned to the hosting NE, to specify whether session data replication is enabled on the NE.
nti	A unique alphanumeric string identifying a network translation index.
defaultTranslationDN	A default E.164 directory number used for translations.
EnterpriseHostsData	An entity that contains data related to enterprise hosts within the Network Server.
HostingNeNodeData_v14_0	An entity that contains data related to an Application Server node or other hosting NE nodes (version 14) within the Network Server.
DefaultSourceIdData	An entity that contains data related to the default source ID within the Network Server.

Element	Description
LinePortData	An entity that contains data related to a line/port within the Network Server.
TrunkGroupData	An entity that contains data related to trunk group within the Network Server.
DataCenterData	An entity that contains data related to the Data Center within the Network Server.
HostingNESetData	An entity that contains data related to a hosting NE set within the Network Server.
EnterpriseData:keepGroupsTogether	Boolean indication (true/false) assigned to <i>EnterpriseData</i> . When set to "true", the group defaults to the preferred node of the enterprise. Otherwise, groups under this enterprise are distributed individually. The group goes to the set linked with the enterprise or the system set if the enterprise has no set.
dnPoolType	Attribute that specifies the pool type associated with a DN.
clusterType	The type of cluster, which can either be <i>primarySecondary</i> (default) or <i>loadBalancing</i> .
userCapacityThousands	This is the total number of users (in thousands) that the cluster can host.
provisioningCapable	When set to "true", the hosting NE is a host that supports provisioning functions for users (such as, an Application Server or Profile Server). When set to "false", the hosting NE does not support provisioning (such as, an Execution Server).
callProcessingCapable	When set to "true", the hosting NE is a host that supports call processing functions for users that it hosts (such as, an Application Server or Execution Server). When set to "false", the hosting NE does not support call processing (such as, Profile Server).
loadBalancingWeight	This is an integer value from 0 through 99. When the node is part of a load-balancing cluster, the greater its weight value, the greater is the number of subscribers that the node can host.
dataCenterId	This is the name of the data center (optional) in which this node belongs.
isOffNet	When set to "false", it indicates that DNs that are "on-net", are assigned and activated.
linePort	The line/port name.
loginIdUrl	The login ID alias for the user who owns the line/port.
newLinePort	The new line/port name.
supportsOrigDiversion	Specifies if a PBX integration line/port is allowed to initiate an originating-diverted call towards the Application Server (also known as a PBX deflection).
TrunkGroupData:name	A unique alphanumeric string identifying a trunk group within the Network Server.
newName	The new name to be specified for a given trunk group within the Network Server
identity	The trunk group identity, including the domain name.

Element	Description
otgDtgIdentity	The trunk group OTG/DTG identity.
HostingNESetData:name	This is the name of the hosting NE set.
systemSet	When set to "true", the hosting NE set is also the system set, otherwise "false".
hostingNEArray	An array of hosting NEs.
DataCenterData:name	The name of the data center.
userAdminArray	An array of UserAdminData
UserAdminData	An entity that contains data related to a hosting NE administrator within the Network Server.
NEMaintenancePartitionData	An entity that contains data related to a NE Maintenance Partition within the Network Server.
NEMaintenancePartitionData:name	A unique alphanumeric string identifying a NE Maintenance Partition within the Network Server.
NEMaintenancePartitionData:systemPartition	Indicates whether this is a system partition.
nodeArray	An array of NodeData.
NodeData	An entity that contains data related to a HostingNe node within the Network Server.
NodeData:name	A unique alphanumeric string identifying a Hosting NE node within the Network Server.

## 4.2 Narrow Results of Get Command

The OSS allows the number of rows returned by a Get command to be narrowed down. To do so, the client provides two attributes: `elementStartIndex` and `numberOfElements`. Attribute `elementStartIndex` specifies the first element that is returned, with index 0 being the first element. Attribute `numberOfElements` specifies how many elements to return, including the element at `elementStartIndex`. These attributes apply to all Get requests.

The maximum number of rows that are returned with one request is 100. If 100 rows are requested by the client or by default, and attributes `elementStartIndex` and `numberOfElements` are incorrect or missing, the following warnings are returned:

- If attribute `elementStartIndex` is missing:  
 "[Warning 16052] elementStartIndex not specified in request. The response contains only the first 100 entries."
- If attribute `numberOfElements` is missing:  
 "[Warning 16053] numberOfElements not specified in request. The response contains only the first 100 entries."
- If both attributes are missing:  
 "[Warning 16051] elementStartIndex and numberOfElements not specified in request. The response contains only the first 100 entries."
- If the `numberOfElements` attribute is set to a value larger than "100" (for example, "110"):

“[Warning 16054] “110” is too large. The response contains a maximum of “100” entries.”

The following examples show usage of these two parameters using the getUserGroups command.

#### 4.2.1 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getUserGroups"
    transactionId="1"
    elementStartIndex="0"
    numberOfElements="2">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

#### 4.2.2 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getUserGroups"
    result="0"
    transactionId="1"
    elementStartIndex="0"
    numberOfElements="2">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.UserGroupData
        enterprise="EnterpriseX"
        groupID="Grp#0"
        groupName="AS1 - Grp#0"
        hostingNE="AS1"
        interLataCarrier="VTL"
        internationalCarrier="VTL"
        intraLataCarrier="VTL"
        profile="Profile2"
        lcaid="grp0_LCA"
        sourceId="4567">
      <dNArray>
        <com.broadsoft.protocols.nsoss.DnData
          dn="12404661234"
          site="DFLT_SITE">
        </com.broadsoft.protocols.nsoss.DnData>
        <com.broadsoft.protocols.nsoss.DnData
          dn="12404661240"
          site="siteA">
        </com.broadsoft.protocols.nsoss.DnData>
        <com.broadsoft.protocols.nsoss.DnData
          dn="12404661250"
          site="siteB">
        </com.broadsoft.protocols.nsoss.DnData>
      </dNArray>
      <dNRangeArray>
        <com.broadsoft.protocols.nsoss.DnRangeData
```

```
        fromDn="12404662000"
        toDn="12404662010"
        site="DFLT_SITE">
</com.broadsoft.protocols.nsoss.DnRangeData>
<com.broadsoft.protocols.nsoss.DnRangeData
  fromDn="12404662020"
  toDn="12404662030"
  site="siteB">
</com.broadsoft.protocols.nsoss.DnRangeData>
</dNRangeArray>
<uRLArray>
  <com.broadsoft.protocols.nsoss.UrlData
    isLoginId="true"
    url="aaaaa@broadsoft.com"
    site="DFLT_SITE">
</com.broadsoft.protocols.nsoss.UrlData>
  <com.broadsoft.protocols.nsoss.UrlData
    isLoginId="true"
    url="bbbbbb@broadsoft.com"
    site="site2">
</com.broadsoft.protocols.nsoss.UrlData>
</uRLArray>
</com.broadsoft.protocols.nsoss.UserGroupData>
<com.broadsoft.protocols.nsoss.UserGroupData
  enterprise="EnterpriseX"
  groupID="GrpEA1"
  groupName="AS1 - GrpEA1"
  hostingNE="AS1"
  interLataCarrier="Bell"
  internationalCarrier="Bell"
  intraLataCarrier="Bell"
  profile="Profile1"
  lcaid="DFLT_LCA_ID"
  sourceId="">
  <dNArray>
  </dNArray>
  <dNRangeArray>
  </dNRangeArray>
  <uRLArray>
  </uRLArray>
</com.broadsoft.protocols.nsoss.UserGroupData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 4.3 Using Authentication, Login, and Logout

Each XML request can send client-specific information within the `com.broadsoft.protocols.nsoss.BroadsoftDocument` element as shown in the following example. The client-specific information contains two optional attributes:

- *loginId*, which specifies the BroadWorks login ID on behalf of which the XML request should be processed.
- *clientId*, which is a client-specific string that can help the client application to correlate requests and responses for different users.

These attributes are optional. If present in an OSS request, the Network Server simply returns them in the corresponding OSS response, without any validation or interpretation. The following shows how the loginId and clientData attributes can be used in any XML OSS request sent to the Network Server.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0" loginId="admin" clientData="OSS Client">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getAllServiceProviderProfiles"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

As of Release 13, the Network Server supports the sharing of an OSS session by multiple login IDs using the authentication, login, and logout mechanism. For this reason, a loginId attribute now also exists in the com.broadsoft.protocols.nsoss.Command element.

When sending OSS commands through the Open Client Server, the BroadWorks Common Communication Transport, the loginId attribute within each command has to be specified so that the Network Server can confirm that the requesting login ID is really authorized and logged in to the system. Otherwise, this loginId attribute is optional and is ignored by the Network Server. The following shows how the loginId should be included in the command element.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getEnterprise"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseData name="Broadsoft">
      </com.broadsoft.protocols.nsoss.EnterpriseData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

The Open Client Server allows clients that connect *via* TCP/IP connections to exchange OSS messages with the Network Server. Only system administrators and provisioning administrators on the Network Server may use the Network Server OSS functions. Each client application must first log in using valid user credentials before performing any other OSS function on the Network Server.

If an error occurs that prevents the Open Client Server from performing a Network Server OSS function, such as a lost connection between the Open Client Server and the Network Server, the Open Client Server sends a Network Server OSS logout message to the client application and ends the session. The client application must then login again; if a client attempts to send another Network Server OSS message without logging in again, the Open Client Server disconnects the client connection immediately.

## 4.3.1 requestAuthentication

### 4.3.1.1 Description

This is the first command that the client application should send as part of the login process. The command verifies if the user is valid and allowed to log in to the Network Server. The Network Server OSS allows a system provider administrator or group provisioning administrator to log in. Upon successful authentication, a nonce value and the encryption algorithm are returned in the response. These two values are used to encrypt the password used in the requestLogin command described in the next command.

#### 4.3.1.1.1 Sample Input XML Structure

```
<?xml version = "1.0" encoding = "ISO-8859-1"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument loginId="admin"
clientData="OSS Client" protocol = "NSOSS" version = "20.0">
  <command commandType = "requestAuthentication">
    <commandData>
      <loginInfo>
        <loginId>admin</loginId>
      </loginInfo>
    </commandData>
  </command>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

#### 4.3.1.1.2 Sample Output XML Structure

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument clientData="OSS Client"
loginId="admin" protocol="NSOSS" version="20.0">
<command commandType="requestAuthentication">
  <result>0</result>
  <commandData>
    <loginInfo>
      <loginId>admin</loginId>
      <nonce>1124742870406</nonce>
      <algorithm>MD5</algorithm>
    </loginInfo>
  </commandData>
</command>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 4.3.2 requestLogin

### 4.3.2.1 Description

This command is used by the client application to log in the user. The Network Server OSS allows a system provider administrator or group provisioning administrator to log in. The *password* field should be encrypted using the user's password, the nonce value, and the algorithm supplied by the Network Server in the above message.

#### Password Calculation

Calculate the message digest of the user's plain password using the Secure Hash Algorithm (SHA) algorithm.

For every 4 bits in the 160-bit digest, starting from the first bit, convert it into a character in ASCII Hex format (0 – 9, a – f). The result is a 40-character string **S1**, for example, f7a9e24777ec23212c54d7a350bc5bea5477fdbb.

Use the string **S1** to construct a new string **S2**: **S2 = nonce + ":" + S1**, where **nonce** is the value in the *requestAuthentication* response message from BroadWorks.

Calculate the message digest of **S2** using the MD5 algorithm.

For every 4 bits in the 128-bit digest from step 2, starting from the first bit, convert it into a character in ASCII Hex format (0 – 9, a – f). The result is a 32-character string, for example, dc70779bf8461b5a1e6aea58f636d5c0.

Use this string as the *password* in the command.

#### 4.3.2.1.1 Sample Input XML Structure

```
<?xml version = "1.0" encoding = "ISO-8859-1"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument loginId="admin"
clientData="OSS Client" protocol = "NSOSS" version = "20.0">
  <command commandType = "requestLogin">
    <commandData>
      <loginInfo>
        <loginId>admin</loginId>
        <password>e26435ce69a9a4569c084564d5cd36e9</password>
      </loginInfo>
    </commandData>
  </command>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

#### 4.3.2.1.2 Sample Output XML Structure

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument clientData="OSS Client"
loginId="admin" protocol="NSOSS" version="20.0">
<command commandType="requestLogin">
  <result>0</result>
  <commandData>
    <loginInfo>
      <loginId>admin</loginId>
      <password>e26435ce69a9a4569c084564d5cd36e9</password>
    </loginInfo>
  </commandData>
</command>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 4.3.3 requestLogout

#### 4.3.3.1 Description

This command is used to log out a user from the Network Server.

#### 4.3.3.1.1 Sample Input XML Structure

```
<?xml version = "1.0" encoding = "ISO-8859-1"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument loginId="admin"
clientData="OSS Client" protocol = "NSOSS" version = "20.0">
  <command commandType = "requestLogout">
```



```
<commandData>
  <loginInfo>
    <loginId>admin</loginId>
    <reason>Logout requested by user</reason>
  </loginInfo>
</commandData>
</command>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

#### 4.3.3.1.2 Sample Output XML Structure

**NOTE:** If the requestLogout request goes through the Open Client Server, the Open Client Server does not return any response back to the client application.

```
<?xml version = "1.0" encoding = "ISO-8859-1"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument loginId="admin"
clientData="OSS Client" protocol = "NSOSS" version = "20.0">
  <command commandType = "requestLogout">
    <result>0</result>
    <commandData>
      <loginInfo>
        <loginId>admin</loginId>
        <reason>Logout requested by user</reason>
      </loginInfo>
    </commandData>
  </command>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5 OSS Commands for System Provisioning

Throughout this section, the sample XML structures contain the `loginId` attribute. This attribute is shown in italics, to indicate that it is only required if the OSS requests are proxied by an Open Client Server.

### 5.1 `getAllServiceProviderProfiles`

#### 5.1.1 Description

This command allows a service provider to retrieve a list of all the public routing profiles currently defined on the Network Server.

**NOTE 1:** Only different profile names are returned and not a list of all policy instances assigned to each profile.

**NOTE 2:** The OSS interface cannot be used to create, modify, or delete a routing profile.

#### 5.1.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getAllServiceProviderProfiles"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

#### 5.1.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getAllServiceProviderProfiles"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.ProfileData
        name="NIL_PROFILE">
      </com.broadsoft.protocols.nsoss.ProfileData>
      <com.broadsoft.protocols.nsoss.ProfileData
        name="Profile1">
      </com.broadsoft.protocols.nsoss.ProfileData>
      <com.broadsoft.protocols.nsoss.ProfileData
        name="Profile2">
      </com.broadsoft.protocols.nsoss.ProfileData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

```
</com.broadsoft.protocols.nsooss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.2 getAllCarriers

### 5.2.1 Description

This command allows a service provider to retrieve a list of all carriers currently defined on the Network Server.

**NOTE 1:** Only different carrier names are returned and not the attributes assigned to each carrier.

**NOTE 2:** The OSS interface cannot be used to create, modify, or delete a carrier.

### 5.2.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="getAllCarriers"  
    loginId="admin"  
    transactionId="1">  
    <commandDataArray>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsooss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.2.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="getAllCarriers"  
    loginId="admin"  
    result="0"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsooss.CarrierData  
        countryCode=""  
        name="NILCAC">  
      </com.broadsoft.protocols.nsooss.CarrierData>  
      <com.broadsoft.protocols.nsooss.CarrierData  
        countryCode="1"  
        name="Bell">  
      </com.broadsoft.protocols.nsooss.CarrierData>  
      <com.broadsoft.protocols.nsooss.CarrierData  
        countryCode="49"  
        name="VTL">  
      </com.broadsoft.protocols.nsooss.CarrierData>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsooss.Command>  
</commandArray>
```

```
</com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.3 getLocalCallingAreas

### 5.3.1 Description

This command allows a service provider to retrieve a list of all local calling areas (LCAs) currently defined on the Network Server.

**NOTE 1:** Only LCA IDs are returned and not the attributes assigned to each LCA.

**NOTE 2:** The OSS interface cannot be used to create, modify, or delete an LCA ID.

### 5.3.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="getLocalCallingAreas"  
    loginId="admin"  
    transactionId="1">  
    <commandDataArray>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.3.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="getLocalCallingAreas"  
    loginId="admin"  
    result="0"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsoss.LCADATA  
        lcaid="DFLT_LCA_ID">  
      </com.broadsoft.protocols.nsoss.LCADATA>  
      <com.broadsoft.protocols.nsoss.LCADATA  
        lcaid="LCA_33">  
      </com.broadsoft.protocols.nsoss.LCADATA>  
      <com.broadsoft.protocols.nsoss.LCADATA  
        lcaid="entX_LCA">  
      </com.broadsoft.protocols.nsoss.LCADATA>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.4 getAllPrivatePolicies

### 5.4.1 Description

This command allows a service provider to retrieve a list of all private routing policies currently defined on the Network Server.

**NOTE 1:** Only different private routing policy names are returned and not details about each private policy instance.

**NOTE 2:** The OSS interface cannot be used to create, modify, or delete a private policy.

### 5.4.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getAllPrivatePolicies"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.4.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getAllPrivatePolicies"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.PrivatePolicyData
        name="voiceVPN">
      </com.broadsoft.protocols.nsooss.PrivatePolicyData>
      <com.broadsoft.protocols.nsooss.PrivatePolicyData
        name="FarEndHopOff">
      </com.broadsoft.protocols.nsooss.PrivatePolicyData>
      <com.broadsoft.protocols.nsooss.PrivatePolicyData
        name="EntMediaSrvSel">
      </com.broadsoft.protocols.nsooss.PrivatePolicyData>
      <com.broadsoft.protocols.nsooss.PrivatePolicyData
        name="EntSubLocation">
      </com.broadsoft.protocols.nsooss.PrivatePolicyData>
      <com.broadsoft.protocols.nsooss.PrivatePolicyData
        name="ExtDialing">
      </com.broadsoft.protocols.nsooss.PrivatePolicyData>
      <com.broadsoft.protocols.nsooss.PrivatePolicyData
        name="EntSIMPLE">
      </com.broadsoft.protocols.nsooss.PrivatePolicyData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

```
<com.broadsoft.protocols.nsooss.PrivatePolicyData
  name="EntNGWRouting">
</com.broadsoft.protocols.nsooss.PrivatePolicyData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.5 addPrivatePolicyInstance

### 5.5.1 Description

This command allows a system provider to create a new private policy instance and to assign it to an enterprise on the Network Server.

- If the “enabled” flag is set to “true”, the instance becomes usable right away.
- If the “enabled” flag is set to “false” or if it is absent, the instance is assigned but remains disabled during route processing.

Except for the EntSubLocation policy, private policy instance attributes other than the “enabled” flag cannot be set or modified through the OSS interface. For EntSubLocation, all the policy instance attributes can be specified using the “com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanceData” element instead of the usual “com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData” element.

**NOTE 1:** The enterprise to which the private policy instance is assigned must exist on the Network Server. An enterprise can only have one instance of a given private policy.

**NOTE 2:** The supported policy names are described in the DTD.

**NOTE 3:** Private policy instance attributes other than the “enabled” flag cannot be set or modified through the OSS interface. For example, the default selector flag (dfltSel) for the private policy voiceVPN can only be changed using the CLI or web interface.

**NOTE 4:** Only one private policy instance can be added per addPrivatePolicyInstance command element. To add additional private policy instances, use additional addPrivatePolicyInstance command elements.

**NOTE 5:** For EntSubLocation, the “intraSiteOnly” attribute can be specified as part of the “com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanceData” element. If not specified, it defaults to “false”.

### 5.5.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addPrivatePolicyInstance"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData
        enabled="true"
        enterprise="EnterpriseX"
        name="voiceVPN">
```

```

        </com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData>
    </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
<com.broadsoft.protocols.nsooss.Command
    commandType="addPrivatePolicyInstance"
    transactionId="2">
    <commandDataArray>
<com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLoc
ationInstanceData
    enabled="true"
    enterprise="EnterpriseX"
    intraSiteOnly="true"
    name="EntSubLocation">
    </
com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLoca
tionInstanceData >
    </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>

```

### 5.5.3 Sample Output XML Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
    <com.broadsoft.protocols.nsooss.Command
        commandType="addPrivatePolicyInstance"
        loginId="admin"
        result="0"
        transactionId="1">
        <commandDataArray>
        </commandDataArray>
    </com.broadsoft.protocols.nsooss.Command>
    <com.broadsoft.protocols.nsooss.Command
        commandType="addPrivatePolicyInstance"
        result="0"
        transactionId="2">
        <commandDataArray>
        </commandDataArray>
    </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>

```

## 5.6 deletePrivatePolicyInstance

### 5.6.1 Description

This command allows a service provider to delete a private policy instance assigned to an enterprise. It deletes the instance and all associated data, such as dial plans. If you want to simply disable the instance, then the modifyPrivatePolicyInstance command should be used instead.

**NOTE 1:** To use this command, the enterprise must exist on the Network Server and the corresponding private policy instance must be assigned.

**NOTE 2:** The supported policy names are described in the DTD.

**NOTE 3:** The enabled flag is ignored.

**NOTE 4:** Only one private policy instance can be deleted per `deletePrivatePolicyInstance` command element. To delete additional private policy instances, use additional `deletePrivatePolicyInstance` command elements.

**NOTE 5:** For the `EntSubLocation` policy, the `"com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanceData"` element can be used instead of the usual `"com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData"`. However, this provides no additional capability as the Network Server simply ignores all policy instance attributes when deleting a policy instance.

### 5.6.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deletePrivatePolicyInstance"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData
        enterprise="EnterpriseX"
        name="voiceVPN">
      </com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deletePrivatePolicyInstance"
    transactionId="2">
    <commandDataArray>
      <com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLoc
ationInstanceData
        enterprise="EnterpriseX"
        name="EntSubLocation">
      </com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLo
cationInstanceData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.6.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deletePrivatePolicyInstance"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
```



```
<com.broadsoft.protocols.nsooss.Command
  commandType="deletePrivatePolicyInstance"
  result="0"
  transactionId="2">
  <commandDataArray>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.7 modifyPrivatePolicyInstance

### 5.7.1 Description

This command allows a service provider to modify certain attributes of a private policy instance assigned to an enterprise. Except for the EntSubLocation policy, the only attribute value that can be changed is the “enabled” status. For EntSubLocation, all the policy instance attributes can be changed using the “com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanceData” element instead of the usual “com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData” element.

**NOTE 1:** To use this command, the enterprise must exist on the Network Server and the corresponding private policy instance must be assigned.

**NOTE 2:** The supported policy names are described in the DTD.

**NOTE 3:** The “enabled” status passed by this command replaces the status stored on the Network Server. If the new status is the same as the one on the Network Server, the command succeeds. If the “enabled” status is not specified in the request, the Network Server disables the instance.

**NOTE 4:** Private policy instance attributes other than the “enabled” flag cannot be set or modified through the OSS interface. For example, the default selector flag (dfltSel) for private policy voiceVPN can only be changed using the CLI or web interface.

**NOTE 5:** Only one private policy instance can be modified per modifyPrivatePolicyInstance command element. To modify additional private policy instances, use additional modifyPrivatePolicyInstance command elements.

**NOTE 6:** For EntSubLocation, the “intraSiteOnly” attribute can be set as part of the “com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanceData” element. If not specified, it remains unchanged on the Network Server.

### 5.7.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyPrivatePolicyInstance"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
```

```

        <com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData
            enabled="true"
            enterprise="EnterpriseX"
            name="voiceVPN">
        </com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="modifyPrivatePolicyInstance"
    transactionId="2">
    <commandDataArray>
<com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLoc
ationInstanceData
    enabled="true"
    enterprise="EnterpriseX"
    intraSiteOnly="true"
    name="EntSubLocation">
</com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLo
cationInstanceData>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

### 5.7.3 Sample Output XML Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
    <com.broadsoft.protocols.nsoss.Command
        commandType="modifyPrivatePolicyInstance"
        loginId="admin"
        result="0"
        transactionId="1">
        <commandDataArray>
        </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
    <com.broadsoft.protocols.nsoss.Command
        commandType="modifyPrivatePolicyInstance"
        result="0"
        transactionId="2">
        <commandDataArray>
        </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

## 5.8 getPrivatePolicyInstance

### 5.8.1 Description

This command allows a service provider to retrieve certain attributes for a private policy instance assigned to an enterprise. Except for the EntSubLocation policy, only the “enabled” status flag is returned with this command. For EntSubLocation, all the policy instance attributes are returned using the “com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanceData” element instead of the usual “com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData” element.

**NOTE 1:** To use this command, the enterprise must exist on the Network Server and the corresponding private policy instance must be assigned.

**NOTE 2:** The supported policy names are described in the DTD.

**NOTE 3:** The “enabled” flag is ignored (if present) in the request and is populated in the response.

**NOTE 4:** Only one private policy instance can be retrieved per `getPrivatePolicyInstance` command element. To retrieve additional private policy instances, use additional `getPrivatePolicyInstance` command elements.

**NOTE 5:** For `EntSubLocation`, the request XML can be built using the “`com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLocationInstanceData`” element or the “`com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData`” element.

### 5.8.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getPrivatePolicyInstance"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData
        enterprise="EnterpriseX"
        name="voiceVPN">
      </com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
<com.broadsoft.protocols.nsooss.Command
  commandType="getPrivatePolicyInstance"
  transactionId="2">
  <commandDataArray>
    <com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData
      enterprise="EnterpriseX"
      name="EntSubLocation">
    </com.broadsoft.protocols.nsooss.PrivatePolicyInstanceData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.8.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getPrivatePolicyInstance"
    result="0"
    loginId="admin"
    transactionId="1">
```

```

<commandDataArray>
  <com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData
    enabled="true"
    enterprise="EnterpriseX"
    name="voiceVPN">
  </com.broadsoft.protocols.nsoss.PrivatePolicyInstanceData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
  commandType="getPrivatePolicyInstance"
  result="0"
  transactionId="2">
  <commandDataArray>
<com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLoc
ationInstanceData
  enabled="true"
  enterprise="EnterpriseX"
  intraSiteOnly="true"
  name="EntSubLocation">
</com.broadsoft.routingserver.policies.privatesubscriberlocation.EntSubLo
cationInstanceData>
  </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

## 5.9 synchronizeEnterprise

### 5.9.1 Description

This command allows a service provider to create an enterprise. If the enterprise exists, the operation has no effect. If the carrier information is not provided, the carrier is set to "NILCAC".

**NOTE 1:** Only one enterprise can be synchronized per synchronizeEnterprise command element. To synchronize additional enterprises, use additional synchronizeEnterprise command elements.

**NOTE 2:** If not specified in the request, call logging defaults to "false" (that is, call logging is disabled) for a new enterprise.

**NOTE 3:** Specifying the privatePoliciesArray element has no effect for this command. To (un)assign private Routing policies, use the addPrivatePolicyInstance or deletePrivatePolicyInstance commands.

**NOTE 4:** If specified, the lcaid must refer to an existing lcaid in the system. The lcaid can also be set to an empty string.

**NOTE 5:** The sourceId is optional. If specified, it is used to associate the enterprise to a source ID or originating trunk group (OTG).

### 5.9.2 Sample Input XML Structure

The example assumes that:

- EnterpriseX and EnterpriseY do not exist in the system and that GrpEA1 exists in the system.

- Carrier “Bell” is provisioned as a valid carrier for intraLATA, interLATA, and international calls.
- Profile “Profile1” is provisioned as a valid public routing profile in the system. If the profile attribute is not specified in the request, the Network Server automatically assigns the system default profile to the new enterprise.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeEnterprise"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseData
        description="Enterprise X"
        name="EnterpriseX"
        interLataCarrier="Bell"
        internationalCarrier="Bell"
        intraLataCarrier="Bell"
        profile="NIL_PROFILE"
        callLogging="true"
        lcaid="DFLT_LCA_ID"
        sourceId="12345">
      </com.broadsoft.protocols.nsoss.EnterpriseData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>

  <com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeEnterprise"
    loginId="admin"
    transactionId="2">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseData
        name="GrpEA1">
      </com.broadsoft.protocols.nsoss.EnterpriseData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>

  <com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeEnterprise"
    loginId="admin"
    transactionId="3">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseData
        description="Enterprise Y"
        name="EnterpriseY"
        interLataCarrier="Bell"
        internationalCarrier="Bell"
        intraLataCarrier="Bell"
        profile="Profile1"
        callLogging="true">
      </com.broadsoft.protocols.nsoss.EnterpriseData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.9.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeEnterprise"
    loginId="admin"
    result="0"
    transactionId="1">
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeEnterprise"
    loginId="admin"
    result="0"
    transactionId="2">
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeEnterprise"
    loginId="admin"
    result="0"
    transactionId="3">
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.10 modifyEnterprise

### 5.10.1 Description

This command allows a service provider to modify an existing enterprise. Although the DTD cannot capture this rule, the `interLataCarrier`, `intraLataCarrier`, `internationalCarrier`, and profile information MUST be provided when modifying the enterprise's properties.

**NOTE 1:** If the description is not provided, the current description is set to "".

**NOTE 2:** If not specified in the request, call logging becomes "false" (that is, call logging is disabled) for the enterprise.

**NOTE 3:** Only one enterprise can be modified per `modifyEnterprise` command element. To modify additional enterprises, use additional `modifyEnterprise` command elements.

**NOTE 4:** Specifying the `privatePoliciesArray` element has no effect for this command. To (un)assign private Routing policies, use the `addPrivatePolicyInstance` or `deletePrivatePolicyInstance` commands.

**NOTE 5:** If specified, the "lcaid" must refer to an existing lcaid in the system. If the "lcaid" is set to an empty string, the enterprise no longer has its own local calling area.

**NOTE 6:** If specified, the "sourceld" is used to associate the enterprise to a source ID, or originating trunk group (OTG). If a sourceld is no longer required, set the "sourceld" to an empty string.

### 5.10.2 Sample Input XML Structure

The following example assumes that:

- Carrier "VTL" is provisioned as a valid carrier for intraLATA, interLATA, and international calls.
- Profile "Profile2" is provisioned as a valid public routing profile in the system.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyEnterprise"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseData
        description="Enterprise X modified"
        name="EnterpriseX"
        interLataCarrier="VTL"
        internationalCarrier="VTL"
        intraLataCarrier="VTL"
        profile="Profile2"
        callLogging="false"
        lcaid="entX_LCA"
        sourceId="76458">
      </com.broadsoft.protocols.nsooss.EnterpriseData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.10.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyEnterprise"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.11 deleteEnterprise

### 5.11.1 Description

This command allows a service provider to delete an existing enterprise. If the enterprise has one or more associated user groups, it cannot be deleted until all user groups are deleted, using individual deleteUserGroup commands. If the enterprise has one or more associated administrators, it cannot be deleted until all the enterprise administrators are deleted, using individual deleteEnterpriseAdmin commands. If the enterprise has one or more associated sites (other than DFLT\_SITE), it cannot be deleted until all the sites are deleted, using individual deleteSite commands. As well, all assigned private policies must be deleted using deletePrivatePolicyInstance commands. In general, an enterprise cannot be deleted until all references to it are first removed from the system.

**NOTE 1:** Only one enterprise can be deleted per deleteEnterprise command element. To delete additional enterprises, use additional deleteEnterprise command elements.

**NOTE 2:** Specifying the privatePoliciesArray element has no effect on this command.

### 5.11.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteEnterprise"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseData
        name="EnterpriseX">
      </com.broadsoft.protocols.nsooss.EnterpriseData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.11.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteEnterprise"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```



## 5.12 getEnterprise

### 5.12.1 Description

This command allows a service provider to query the Network Server for an existing enterprise. The list of assigned private Routing policies is provided in the `privatePoliciesArray` element.

**NOTE:** Only one enterprise can be queried for per `getEnterprise` command element. To query for additional enterprises, use additional `getEnterprise` command elements.

### 5.12.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getEnterprise"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseData
        name="EnterpriseX">
      </com.broadsoft.protocols.nsooss.EnterpriseData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.12.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getEnterprise"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseData
        description="Enterprise X modified"
        name="EnterpriseX"
        interLataCarrier="VTL"
        internationalCarrier="VTL"
        intraLataCarrier="VTL"
        profile="Profile2"
        callLogging="false"
        lcaid="entX_LCA"
        sourceId="76458"
        sourceIdForEAPrefix="EnterpriseXCicSourceId">
      <privatePoliciesArray>
        <string
          value="voiceVPN">
        </string>
        <string
          value="FarEndHopOff">
```

```
        </string>
      </privatePoliciesArray>
    </com.broadsoft.protocols.nsooss.EnterpriseData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.13 getEnterprises

### 5.13.1 Description

This command allows a service provider to query the Network Server system for all existing enterprises. The list of assigned private Routing policies is provided in the `privatePoliciesArray` element for each enterprise.

**NOTE:** `NIL_ENTERPRISE` is a system-defined entity that cannot be modified or deleted. User groups and hosting network elements can have `NIL_ENTERPRISE` as their designated enterprise.

### 5.13.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getEnterprises"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.13.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getEnterprises"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseData
        description="Default enterprise"
        name="NIL_ENTERPRISE"
        interLataCarrier="NILCAC"
        internationalCarrier="NILCAC"
        intraLataCarrier="NILCAC"
        profile="NIL_PROFILE"
        callLogging="false"
        lcaid=""
        sourceId=""
        sourceIdForEAPrefix="">>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

```
<privatePoliciesArray>
</privatePoliciesArray>
</com.broadsoft.protocols.nsoss.EnterpriseData>
<com.broadsoft.protocols.nsoss.EnterpriseData
  description="EA CAC for each type (intra,inter,intl)"
  name="GrpEA1"
  interLataCarrier="Bell"
  internationalCarrier="Bell"
  intraLataCarrier="Bell"
  profile="Profile1"
  callLogging="false"
  lcaid="DFLT_LCA_ID"
  sourceId=""
  sourceIdForEAPrefix=" ">
  <privatePoliciesArray>
    <string
      value="voiceVPN">
    </string>
  </privatePoliciesArray>
</com.broadsoft.protocols.nsoss.EnterpriseData>
<com.broadsoft.protocols.nsoss.EnterpriseData
  description="Enterprise X modified"
  name="EnterpriseX"
  interLataCarrier="VTL"
  internationalCarrier="VTL"
  intraLataCarrier="VTL"
  profile="Profile2"
  callLogging="false"
  lcaid="entX_LCA"
  sourceId="76458"
  sourceIdForEAPrefix="EnterpriseXCicSourceId">>
  <privatePoliciesArray>
    <string
      value="voiceVPN">
    </string>
    <string
      value="FarEndHopOff">
    </string>
  </privatePoliciesArray>
  </com.broadsoft.protocols.nsoss.EnterpriseData>
</commanddataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.14 getHostsForEnterprise

### 5.14.1 Description

This command allows a service provider to query the Network Server to get a list of hostingNe that hosts a given enterprise.

**NOTE:** NIL\_ENTERPRISE is a system-defined entity that cannot be modified or deleted. User groups and hosting network elements can have NIL\_ENTERPRISE as their designated enterprise.

## 5.14.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>

  <com.broadsoft.protocols.nsoss.Command
    commandType="getHostsForEnterprise"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseHostsData
        enterpriseName="Broadsoft">
      </com.broadsoft.protocols.nsoss.EnterpriseHostsData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.14.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getHostsForEnterprise"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseHostsData
        enterpriseName="Broadsoft">
        <hostsArray>
          <string value="ASaus"></string>
          <string value="ASsanity"></string>
        </hostsArray>
      </com.broadsoft.protocols.nsoss.EnterpriseHostsData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.15 addSite

### 5.15.1 Description

This command allows a service provider to create an enterprise site. If the enterprise exists, the operation has no effect. If carrier information is not provided, the carrier is set to "NILCAC".

**NOTE 1:** This command can only be used if the target enterprise exists in the system.

**NOTE 2:** The "name" attribute is required in the command to specify the name of the new site. The command fails if the name is already specified for an existing site in the enterprise.

**NOTE 3:** The “description” attribute is optional. If not specified, the server sets a blank description (“”) for the new site.

**NOTE 4:** The “profile” attribute is optional. If specified, it must refer to an existing routing profile in the system. Otherwise, the server assigns “NIL\_PROFILE” to the new site.

**NOTE 5:** The “sourceld” is optional. If specified, it is used to associate the site to a source ID, or originating trunk group (OTG).

**NOTE 6:** Only one site can be added per addSite command element. To add additional sites, use additional addSite command elements.

### 5.15.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addSite"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.SiteData
        enterprise="EnterpriseX"
        name="siteA"
        description=""
        profile="Profall"
        sourceId="12345">
      </com.broadsoft.protocols.nsoss.SiteData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.15.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addSite"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.16 modifySite

### 5.16.1 Description

This command allows a service provider to modify an existing site. The unique key of the site (name and enterprise attributes) is required and must refer to an existing site in the system.

**NOTE 1:** If the description is not provided, the current description is set to "".

**NOTE 2:** If the profile is not provided, the server assigns "NIL\_PROFILE" to the site.

**NOTE 3:** If specified, the "sourceld" is used to associate the site to a source ID, or originating trunk group (OTG). If a sourceld is no longer required, set the "sourceld" to an empty string. The sourceld remains unchanged if the attribute is not specified in the request.

**NOTE 4:** Only one site can be modified per modifySite command element. To modify additional sites, use additional modifySite command elements.

### 5.16.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifySite"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.SiteData
        enterprise="EnterpriseX"
        name="siteA"
        description="Site A"
        profile="Profile1"
        sourceId="76458">
      </com.broadsoft.protocols.nsoss.SiteData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.16.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifySite"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
```

```
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.17 deleteSite

### 5.17.1 Description

This command allows a service provider to delete an existing site. If the site is referenced in the system, it cannot be deleted. The unique key of the site (name and enterprise attributes) is required and must refer to an existing site in the system. The DFLT\_SITE of an enterprise cannot be deleted.

**NOTE:** Only one site can be deleted per deleteSite command element. To delete additional sites, use additional deleteSite command elements.

### 5.17.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteSite"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.SiteData
        enterprise="EnterpriseX"
        name="siteA"
      </com.broadsoft.protocols.nsoss.SiteData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.17.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteSite"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.18 getSite

### 5.18.1 Description

This command allows a service provider to query the Network Server for an existing site.

**NOTE:** Only one site can be queried for per getSite command element. To query for additional sites, use additional getSite command elements.

### 5.18.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getSite"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.SiteData
        enterprise="EnterpriseX"
        name="siteA"
      </com.broadsoft.protocols.nsoss.SiteData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.18.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getSite"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.SiteData
        description="Site A"
        enterprise="EnterpriseX"
        name="siteA"
        profile="Profile1"
        sourceId="76458">
      </com.broadsoft.protocols.nsoss.SiteData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```



## 5.19 getSites

### 5.19.1 Description

This command allows a service provider to query the Network Server system for all existing sites.

### 5.19.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getSites"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.19.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getSites"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.SiteData
        description="Site A"
        enterprise="EnterpriseX"
        name="siteA"
        profile="Profile1"
        sourceId="76458">
      </com.broadsoft.protocols.nsooss.SiteData>
      <com.broadsoft.protocols.nsooss.SiteData
        description="Site B"
        enterprise="EnterpriseX"
        name="siteB"
        profile="NIL_PROFILE"
        sourceId="">
      </com.broadsoft.protocols.nsooss.SiteData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.20 addHostingNe

### 5.20.1 Description

This command allows a service provider to create a new hosting NE in the system. The hosting NE name cannot be used by another hosting NE in the system. Although the DTD cannot capture this rule, the profile information must be provided when adding a new hosting NE.

**NOTE 1:** If the “defaultSynchEnterprise” attribute is specified in the request, it must refer to a valid enterprise provisioned in the system. If it is not specified, the Network Server assigns NIL\_ENTERPRISE as the default synchronization enterprise for the new hosting NE.

**NOTE 2:** If the “defaultSynchSite” attribute is specified in the request, it must refer to a valid site provisioned in the enterprise identified by the “defaultSynchEnterprise” attribute. If it is not specified, the Network Server assigns DFLT\_SITE as the default synchronization site for the new hosting NE.

**NOTE 3:** If the “defaultRoutingEnterprise” attribute is specified in the request, it must refer to a valid enterprise provisioned in the system. If it is not specified, the Network Server assigns NIL\_ENTERPRISE as the default routing enterprise for the new hosting NE.

**NOTE 4:** If the “defaultRoutingSite” attribute is specified in the request, it must refer to a valid site provisioned in the enterprise identified by the “defaultRoutingEnterprise” attribute. If it is not specified, the Network Server assigns DFLT\_SITE as the default routing site for the new hosting NE.

**NOTE 5:** If the “supportsE164” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not support E164 directory number format.

**NOTE 6:** If the “isCallTypeInfoRequired” attribute is not specified in the request, the Network Server assumes that the new hosting NE should not receive call type and call category information transmitted by the Network Server.

**NOTE 7:** If the “supportsPhoneContext” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not support the phone context in SIP messages.

**NOTE 8:** If the “supportsForking” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not support parallel sending of SIP messages.

**NOTE 9:** If the “supportsSourceId” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not support the originating trunk group (OTG) functionality.

**NOTE 10:** If the “countryCode” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not have a default country code. If specified, the country code must refer to a valid country code entry provisioned in the system.

**NOTE 11:** If the “state” attribute is not specified in the request, the Network Server assumes that the new hosting NE should be in state *OffLine*. If specified, the state must be either *OffLine* or *OnLine*.

**NOTE 12:** If the “type” attribute is not specified in the request, the Network Server assumes that the new hosting NE is of type “other”. If specified, the type must be either “broadworks” (if the hosting NE represents an Application Server) or “other”.

**NOTE 13:** If the “poll” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not require polling. If specified, the poll value must be either “true” (to enable polling) or “false”.

**NOTE 14:** If the “signalCicAddedAsCacToAs” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not send the CIC as CAC to the Application Server. If specified, the signalCicAddedAsCacToAs value must be either “true” or “false”.

**NOTE 15:** Only one hosting NE can be added per addHostingNe command element. To add additional hosting NEs, use additional addHostingNe command elements.

### 5.2.0.2 Sample Input XML Structure

The following example assumes that profile “Profile1” is provisioned as a valid public routing profile in the system.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addHostingNe"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNeData
        countryCode="1"
        defaultRoutingEnterprise="NIL_ENTERPRISE"
        defaultRoutingSite="DFLT_SITE"
        defaultSynchEnterprise="NIL_ENTERPRISE"
        defaultSynchSite="DFLT_SITE"
        name="AS1"
        poll="false"
        profile="Profile1"
        state="OnLine"
        supportsE164="true"
        isCallTypeInfoRequired="true"
        supportsPhoneContext="false"
        supportsForking="false"
        supportsSourceId="true"
        isNSReasonSupported="true"
        isSubLocationRoutingBypass="false"
        isESLCallCategoryOverride="false"
        requiresChargeIndication="false"
        requiresE164Number="true"
        xspVersionEqual="true"
        sessionReplicationEnabled="true"
        type="broadworks"
        clusterType="primarySecondary"
        userCapacityThousands="500"
        provisioningCapable="true"
        callProcessingCapable="true"
        sipIpv4NotSupported="false"
        sipIpv6NotSupported="false"
        signalCicAddedAsCacToAS="false">
      </com.broadsoft.protocols.nsooss.HostingNeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

```
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.20.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addHostingNe"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.21 deleteHostingNe

### 5.21.1 Description

This command allows a service provider to delete an existing hosting network element from the system. To use this command the hosting NE must be defined in the system and no references to this hosting NE should exist, such as in user groups and alternate addresses.

**NOTE 1:** All attributes other than "name" specified in the request are ignored.

**NOTE 2:** Only one hosting NE can be deleted per deleteHostingNe command element. To delete additional hosting NEs, use additional deleteHostingNe command elements.

### 5.21.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteHostingNe"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNeData
        name="AS1">
      </com.broadsoft.protocols.nsooss.HostingNeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.21.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteHostingNe"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.22 modifyHostingNe

### 5.22.1 Description

This command allows a service provider to modify the properties of an existing hosting network element defined in the system. Although the DTD cannot capture this rule, the profile information must be provided when modifying a hosting NE.

**NOTE 1:** If the “defaultSynchEnterprise” attribute is specified in the request, it must refer to a valid enterprise provisioned in the system. When not specified, the Network Server assigns NIL\_ENTERPRISE as the default synchronization enterprise for the hosting NE.

**NOTE 2:** If the “defaultSynchSite” attribute is specified in the request, it must refer to a valid site provisioned in the enterprise identified by the “defaultSynchEnterprise” attribute. If it is not specified, the Network Server assigns DFLT\_SITE as the default synchronization site for the hosting NE.

**NOTE 3:** If the “defaultRoutingEnterprise” attribute is specified in the request, it must refer to a valid enterprise provisioned in the system. If it is not specified, the Network Server assigns NIL\_ENTERPRISE as the default routing enterprise for the hosting NE.

**NOTE 4:** If the “defaultRoutingSite” attribute is specified in the request, it must refer to a valid site provisioned in the enterprise identified by the “defaultRoutingEnterprise” attribute. If it is not specified, the Network Server assigns DFLT\_SITE as the default routing site for the hosting NE.

**NOTE 5:** If the “supportsE164” attribute is not specified in the request, the Network Server assumes that the hosting NE does not support E164 directory number format.

**NOTE 6:** If the “isCallTypeInfoRequired” attribute is not specified in the request, the Network Server assumes that the hosting NE should not receive call type and call category information transmitted by the Network Server.

**NOTE 7:** If the “supportsPhoneContext” attribute is not specified in the request, the Network Server assumes that the hosting NE does not support the phone context in SIP messages.

**NOTE 8:** If the “supportsForking” attribute is not specified in the request, the Network Server assumes that the hosting NE does not support parallel sending of SIP messages.

**NOTE 9:** If the “supportsSourceId” attribute is not specified in the request, the Network Server assumes that the hosting NE does not support the originating trunk group (OTG) functionality.

**NOTE 10:** If the “countryCode” attribute is not specified in the request, the Network Server assumes that the hosting NE does not have a default country code. If specified, the country code must refer to a valid country code entry provisioned in the system.

**NOTE 11:** If the “state” attribute is not specified in the request, the Network Server assumes that the hosting NE should remain in its current state. If specified, the state must be either *OffLine* or *OnLine*.

**NOTE 12:** If the “type” attribute is not specified in the request, the Network Server assumes that the hosting NE should keep its current type. If specified, the type must be either “broadworks” (if the hosting NE represents an Application Server) or “other”.

**NOTE 13:** If the “poll” attribute is not specified in the request, the Network Server assumes that the hosting NE does not require polling. If specified, the poll value must be either “true” (to enable polling) or “false”.

**NOTE 14:** If the “signalCicAddedAsCacToAs” attribute is not specified in the request, the Network Server assumes that the new hosting NE does not send the CIC as CAC to the Application Server. If specified, the signalCicAddedAsCacToAs value must be either “true” or “false”.

**NOTE 15:** Only one hosting NE can be modified per modifyHostingNe command element. To modify additional hosting NEs, use additional modifyHostingNe command elements.

### 5.22.2 Sample Input XML Structure

The following example assumes that profile “Profile1” is provisioned as a valid public routing profile in the system.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyHostingNe"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeData
        countryCode="1"
        defaultRoutingEnterprise="EnterpriseX"
        defaultRoutingSite="SiteA"
        defaultSynchEnterprise="EnterpriseX"
        defaultSynchSite="SiteA"
        name="AS1"
        poll="true"
        profile="Profile1"
        state="OffLine"
        supportsE164="true"
        isCallTypeInfoRequired="false"
        supportsPhoneContext="true"
        supportsForking="true"
        supportsSourceId="false"
        isNSReasonSupported="true"
```

```
isSubLocationRoutingBypass="false"
isESLCallCategoryOverride="false"
requiresChargeIndication="false"
requiresE164Number="true"
xspVersionEqual="true"
sessionReplicationEnabled="true"
type="other"
clusterType="primarySecondary"
userCapacityThousands="500"
provisioningCapable="true"
callProcessingCapable="true"
sipIpv4NotSupported="false"
sipIpv6NotSupported="false"
signalCicAddedAsCacToAS="false">
</com.broadsoft.protocols.nsoss.HostingNeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.22.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="modifyHostingNe"
loginId="admin"
result="0"
transactionId="1">
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument
```

## 5.23 getHostingNes

### 5.23.1 Description

This command allows a service provider to retrieve the properties of all existing hosting network elements defined in the system.

### 5.23.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="getHostingNe"
loginId="admin"
transactionId="1">
<commandDataArray>
<com.broadsoft.protocols.nsoss.HostingNeData
name="AS1">
</com.broadsoft.protocols.nsoss.HostingNeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
```

```
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.23.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="getHostingNe"  
    loginId="admin"  
    result="0"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsooss.HostingNeData  
        countryCode="1"  
        defaultRoutingEnterprise="NIL_ENTERPRISE"  
        defaultRoutingSite="DFLT_SITE"  
        defaultSynchEnterprise="NIL_ENTERPRISE"  
        defaultSynchSite="DFLT_SITE"  
        name="AS1"  
        poll="true"  
        profile="Profile1"  
        state="OnLine"  
        supportsE164="true"  
        isCallTypeInfoRequired="false"  
        supportsPhoneContext="true"  
        supportsForking="false"  
        supportsSourceId="false"  
        isNSReasonSupported="true"  
        isSubLocationRoutingBypass="false"  
        isESLCallCategoryOverride="false"  
        requiresChargeIndication="false"  
        requiresE164Number="true"  
        xspVersionEqual="true"  
        sessionReplicationEnabled="true"  
        type="other"  
        clusterType="primarySecondary"  
        userCapacityThousands="500"  
        provisioningCapable="true"  
        callProcessingCapable="true"  
        sipIpv4NotSupported="true"  
        sipIpv6NotSupported="true"  
        signalCicAddedAsCacToAS="true">  
      </com.broadsoft.protocols.nsooss.HostingNeData>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsooss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```



## 5.24 getHostingNe

### 5.24.1 Description

This command allows a service provider to retrieve the properties of an existing hosting network element defined in the system. The name of the hosting NE is required in the request and all other attributes are ignored when specified in the input XML string.

**NOTE:** Only one hosting NE can be retrieved per getHostingNe command element. To retrieve additional hosting NEs, use additional getHostingNe command elements.

### 5.24.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getHostingNe"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeData
        name="AS1">
      </com.broadsoft.protocols.nsoss.HostingNeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.24.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getHostingNe"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeData
        countryCode="1"
        defaultRoutingEnterprise="NIL_ENTERPRISE"
        defaultRoutingSite="DFLT_SITE"
        defaultSynchEnterprise="NIL_ENTERPRISE"
        defaultSynchSite="DFLT_SITE"
        name="AS1"
        poll="true"
        profile="Profile1"
        state="OnLine"
        supportsE164="true"
        isCallTypeInfoRequired="false"
        supportsPhoneContext="true"
        supportsForking="false"
        supportsSourceId="false"
        isNSReasonSupported="true"
        isSubLocationRoutingBypass="false"
```

```
isESLCallCategoryOverride="false"
requiresChargeIndication="false"
requiresE164Number="true"
xspVersionEqual="true"
sessionReplicationEnabled="true"
type="other"
clusterType="primarySecondary"
userCapacityThousands="500"
provisioningCapable="true"
callProcessingCapable="true"
sipIpv4NotSupported="true"
sipIpv6NotSupported="true"
signalCicAddedAsCacToAS="true">
  </com.broadsoft.protocols.nsoss.HostingNeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.25 addHostingNeNode

### 5.25.1 Description

This command allows a service provider to define a new node for an existing hosting network element in the system.

**NOTE 1:** Node 0 is automatically created when provisioning a new hosting NE.

**NOTE 2:** The “name” must refer to a valid hosting NE provisioned in the system.

**NOTE 3:** The “nodeId” must not be used by an existing node on the hosting NE. The “nodeId” is a number between 0 and 9.

**NOTE 4:** If the “description” attribute is not specified in the request, the Network Server assumes that the new hosting NE node does not have a description.

**NOTE 5:** If the “isDefault” attribute is not specified in the request, the Network Server assumes that the new hosting NE node is not the default node for new DNS and URLs assigned on this hosting NE.

**NOTE 6:** If the field *version*, *patchLevel*, and *minimumPatchLevel* are not specified in the request, the Network Server initializes them to <unspecified>. These fields are updated the next time the hosting NE node tries to connect to the Network Server through the location API.

**NOTE 7:** Only one hosting NE node can be added per addHostingNeNode command element. To add additional hosting NE nodes, use additional addHostingNeNode command elements.

### 5.25.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addHostingNeNode"
    loginId="admin"
```

```
transactionId="1">
  <commanddataArray>
    <com.broadsoft.protocols.nsoss.HostingNeNodeData
      name="AS1"
      nodeId="1"
      description="AS1 node 1"
      isDefault="false">
    </com.broadsoft.protocols.nsoss.HostingNeNodeData>
  </commanddataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.25.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addHostingNeNode"
    loginId="admin"
    result="0"
    transactionId="1">
    <commanddataArray>
    </commanddataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.26 deleteHostingNeNode

### 5.26.1 Description

This command allows a service provider to delete an existing hosting network element node from the system. To use this command, the hosting NE node must be defined in the system.

**NOTE 1:** Attributes other than "name" and "nodeId" are ignored if specified in the request.

**NOTE 2:** A node flagged as "default" cannot be deleted.

**NOTE 3:** Only one hosting NE node can be deleted per deleteHostingNeNode command element. To delete additional hosting NE nodes, use additional deleteHostingNeNode command elements.

### 5.26.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteHostingNeNode"
    loginId="admin"
    transactionId="1">
```

```
<commanddataArray>
  <com.broadsoft.protocols.nsoss.HostingNeNodeData
    name="AS1"
    nodeId="1">
  </com.broadsoft.protocols.nsoss.HostingNeNodeData>
</commanddataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.26.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteHostingNeNode"
    loginId="admin"
    result="0"
    transactionId="1">
    <commanddataArray>
    </commanddataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.27 modifyHostingNeNode

### 5.27.1 Description

This command allows a service provider to modify an existing hosting network element node defined in the system. The hosting NE node must be defined in the system.

**NOTE 1:** If the "description" attribute is not specified in the request, the Network Server assumes that the hosting NE node does not have a description.

**NOTE 2:** If the "isDefault" attribute is not specified in the request, the Network Server assumes that the hosting NE node is not the default node for new DNS and URLs assigned on this hosting NE. Only one node per hosting NE can be the default at a time.

**NOTE 3:** If the field *version*, *patchLevel*, and *minimumPatchLevel* are not specified in the request, the Network Server initializes them to <unspecified>. These fields are updated the next time the hosting NE node tries to connect to the Network Server through the location API.

**NOTE 4:** Only one hosting NE node can be modified per modifyHostingNeNode command element. To modify additional hosting NE nodes, use additional modifyHostingNeNode command elements.

### 5.27.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
```

```
commandType="modifyHostingNeNode"
loginId="admin"
transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsooss.HostingNeNodeData
      name="AS1"
      nodeId="0"
      description="AS1 node 0"
      isDefault="true">
    </com.broadsoft.protocols.nsooss.HostingNeNodeData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.27.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyHostingNeNode"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.28 getHostingNeNodes

### 5.28.1 Description

This command allows a service provider to retrieve all existing hosting network element nodes defined in the system. The addresses defined for each node are also returned.

### 5.28.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getHostingNeNodes"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.28.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getHostingNeNodes"
    loginId="admin"
    result="0"
    transactionId="1">
    <commanddataArray>
      <com.broadsoft.protocols.nsooss.HostingNeNodeData
        description="AS1 node 0"
        isDefault="true"
        minimumPatchLevel="MP3"
        name="AS1"
        nodeId="0"
        patchLevel="MP3"
        version="Rel1.1_1.131">
        <addressArray>
          <com.broadsoft.protocols.nsooss.HostingNeAddrData
            address="192.168.12.5"
            cost="1"
            port="5060"
            transport="tcp"
            type="DualRouting"
            weight="99">
          </com.broadsoft.protocols.nsooss.HostingNeAddrData>
        </addressArray>
      </com.broadsoft.protocols.nsooss.HostingNeNodeData>
      <com.broadsoft.protocols.nsooss.HostingNeNodeData
        description="AS1 node 1"
        isDefault="false"
        minimumPatchLevel="MP3"
        name="AS1"
        nodeId="1"
        patchLevel="MP3"
        version="Rel1.1_1.131">
        <addressArray>
        </addressArray>
      </com.broadsoft.protocols.nsooss.HostingNeNodeData>
      <com.broadsoft.protocols.nsooss.HostingNeNodeData
        description="Test host node 0"
        isDefault="false"
        minimumPatchLevel="MP3"
        name="AS2"
        nodeId="0"
        patchLevel="MP3"
        version="Rel1.1_1.131">
        <addressArray>
          <com.broadsoft.protocols.nsooss.HostingNeAddrData
            address="192.168.13.4"
            cost="1"
            port="5060"
            transport="tcp"
            type="DualRouting"
            weight="99">
          </com.broadsoft.protocols.nsooss.HostingNeAddrData>
        </addressArray>
      </com.broadsoft.protocols.nsooss.HostingNeNodeData>
    </commanddataArray>
  </com.broadsoft.protocols.nsooss.Command>
```

```
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.29 getHostingNeNode

### 5.29.1 Description

This command allows a service provider to retrieve the properties of an existing hosting network element node defined in the system. The “name” and “nodeId” of the hosting NE node are required in the request and all other attributes are ignored when specified in the input XML string.

**NOTE:** Only one hosting NE node can be retrieved per getHostingNeNode command element. To retrieve additional hosting NE nodes, use additional getHostingNeNode command elements.

### 5.29.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="getHostingNeNode"  
    loginId="admin"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsooss.HostingNeNodeData  
        name="AS1"  
        nodeId="0">  
      </com.broadsoft.protocols.nsooss.HostingNeNodeData>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsooss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.29.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="getHostingNeNode"  
    loginId="admin"  
    result="0"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsooss.HostingNeNodeData  
        description="AS1 node 0"  
        isDefault="true"  
        minimumPatchLevel="MP3"  
        name="AS1"  
        nodeId="0"  
        patchLevel="MP3"  
        version="Rel11.1_1.131">  
      <addressArray>  
        <com.broadsoft.protocols.nsooss.HostingNeAddrData
```

```
        address="192.168.12.5"
        cost="1"
        port="5060"
        transport="tcp"
        type="DualRouting"
        weight="99">
    </com.broadsoft.protocols.nsoss.HostingNeAddrData>
</addressArray>
</com.broadsoft.protocols.nsoss.HostingNeNodeData>
</commanddataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.30 addHostingNeAddr

### 5.30.1 Description

This command allows a service provider to define a new address for an existing hosting network element node in the system.

**NOTE 1:** The “name” and “nodeld” must refer to a valid hosting NE node provisioned in the system.

**NOTE 2:** Although the DTD cannot capture this rule, the “cost” is required and must be provided. Valid costs range from 1 through 9.

**NOTE 3:** If the “port” attribute is not specified in the request, the Network Server assumes that the new address has a port of 5060. If specified, a valid port ranges from 1024 to 65535. The value “-1” is also supported, meaning that the port is not included in the list of contacts returned by the Network Server.

**NOTE 4:** Although the DTD cannot capture this rule, the “type” of address is required and must be provided. Valid types are defined in the DTD. If the type is “Alias”, then the attributes “cost”, “weight”, and “port” are ignored by the Network Server if they are specified in the XML request. The Network Server uses predefined values for cost, weight, and port when the address type is set to “Alias”.

**NOTE 5:** If the “weight” attribute is not specified in the request, the Network Server assumes that the new address has a weight of 0. If specified, a valid weight ranges from 0 through 99.

**NOTE 6:** If the “transport” attribute is not specified in the request, the Network Server assumes that the new address has a transport set to “unspecified”. Valid types are defined in the DTD.

**NOTE 7:** Only one hosting NE node address can be added per addHostingNeAddr command element. To add additional hosting NE node addresses, use additional addHostingNeAddr command elements.

### 5.30.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
```



```
commandType="addHostingNeAddr"
loginId="admin"
transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsoss.HostingNeNodeData
      name="AS1"
      nodeId="0">
      <addressArray>
        <com.broadsoft.protocols.nsoss.HostingNeAddrData
          address="AS1.0.addr"
          cost="1"
          port="5060"
          type="DualRouting"
          weight="50"
          transport="tcp">
        </com.broadsoft.protocols.nsoss.HostingNeAddrData>
      </addressArray>
    </com.broadsoft.protocols.nsoss.HostingNeNodeData>
  </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.30.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addHostingNeAddr"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.31 deleteHostingNeAddr

### 5.31.1 Description

This command allows a service provider to delete an existing hosting network element node address from the system. To use this command, the hosting NE node address must be defined in the system.

**NOTE 1:** Attributes other than "name", "nodeId", and "address" are ignored if specified in the request.

**NOTE 2:** Only one hosting NE node address can be deleted per deleteHostingNeAddr command element. To delete additional hosting NE node addresses, use additional deleteHostingNeAddr command elements.

### 5.31.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteHostingNeAddr"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNeNodeData
        name="AS1"
        nodeId="0">
        <addressArray>
          <com.broadsoft.protocols.nsooss.HostingNeAddrData
            address="AS1.0.addr">
          </com.broadsoft.protocols.nsooss.HostingNeAddrData>
        </addressArray>
      </com.broadsoft.protocols.nsooss.HostingNeNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.31.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteHostingNeAddr"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.32 modifyHostingNeAddr

### 5.32.1 Description

This command allows a service provider to modify an existing hosting network element node address defined in the system. To use this command, the hosting NE node address must be defined in the system.

**NOTE 1:** Although the DTD cannot capture this rule, the “cost” is required and must be provided. Valid costs range from 1 through 9.

**NOTE 2:** If the “port” attribute is not specified in the request, the Network Server assumes that the address has a port of 5060. If specified, a valid port ranges from 1024 to 65535. The value “-1” is also supported, meaning that the port is not included in the list of contacts returned by the Network Server.

**NOTE 3:** Although the DTD cannot capture this rule, the “type” of address is required and must be provided. Valid types are defined in the DTD. If the type is “Alias”, then attributes “cost”, “weight”, and “port” are ignored by the Network Server if they are specified in the XML request. The Network Server uses predefined values for cost, weight, and port when the address type is set to “Alias”.

**NOTE 4:** If the “weight” attribute is not specified in the request, the Network Server assumes that the address has a weight of 0. If specified, a valid weight ranges from 0 through 99.

**NOTE 5:** If the “transport” attribute is not specified in the request, the Network Server assumes that the new address has a transport set to “unspecified”. Valid types are defined in the DTD.

**NOTE 6:** Only one hosting NE node address can be modified per modifyHostingNeAddr command element. To modify additional hosting NE node addresses, use additional modifyHostingNeAddr command elements.

### 5.32.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.
  </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.Broadsoft
broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyHostingNeAddr"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeNodeData
        name="AS1"
        nodeId="0">
        <addressArray>
          <com.broadsoft.protocols.nsoss.HostingNeAddrData
            address="192.168.12.5"
            cost="2"
            port="6000"
            type="Signaling"
            weight="60"
            transport="udp">
          </com.broadsoft.protocols.nsoss.HostingNeAddrData>
        </addressArray>
      </com.broadsoft.protocols.nsoss.HostingNeNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.32.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
```

```
<com.broadsoft.protocols.nsooss.Command
  commandType="modifyHostingNeAddr"
  loginId="admin"
  result="0"
  transactionId="1">
  <commandDataArray>
```

## 5.33 getHostingNeAddr

### 5.33.1 Description

This command allows a service provider to retrieve all addresses defined for an existing hosting network element node.

### 5.33.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getHostingNeAddr"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNeNodeData
        name="AS1"
        nodeId="0">
      </com.broadsoft.protocols.nsooss.HostingNeNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.33.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getHostingNeAddr"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNeNodeData
        description="Test host node 0"
        isDefault="false"
        minimumPatchLevel="mp0"
        name="ASlab"
        nodeId="1"
        patchLevel="mp0"
        version="AS_Rel_18.0_1.378">
      <addressArray>
        <com.broadsoft.protocols.nsooss.HostingNeAddrData
          address="192.168.12.5"
          cost="2"
          port="6000"
          transport="unspecified">
```

```
        type="Signaling"
        weight="0">
        </com.broadsoft.protocols.nsoss.HostingNeAddrData>
    </addressArray>
</com.broadsoft.protocols.nsoss.HostingNeNodeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.34 addHostingNEAdministrators

### 5.34.1 Description

Although hosting network element administrator login IDs are supposed to be managed by automatic synchronization with the Application Servers connected to the Network Server, it may be necessary to make corrections to the Network Server data under abnormal circumstances. This command allows a service provider to add one or more administrator login IDs to a hosting network element.

### 5.34.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addHostingNEAdministrators"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0
        hostingNE="PSecho">
        <userAdminArray>
          <com.broadsoft.protocols.nsoss.UserAdminData
            userId="jsmith@telco.net"
            groupId="winnipeg">
          </com.broadsoft.protocols.nsoss.UserAdminData>
        </userAdminArray>
      </com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.34.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addHostingNEAdministrators"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.35 deleteHostingNEAdministrators

### 5.35.1 Description

Although hosting network element administrator login IDs are supposed to be managed by automatic synchronization with the Application Servers connected to the Network Server, it may be necessary to make corrections to the Network Server data under abnormal circumstances. This command allows a service provider to delete one or more administrator login IDs from a hosting network element.

### 5.35.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteHostingNEAdministrators"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0
        hostingNE="PSecho">
        <userAdminArray>
          <com.broadsoft.protocols.nsoss.UserAdminData
            userId="jsmith@telco.net"
            groupId="winnipeg">
          </com.broadsoft.protocols.nsoss.UserAdminData>
        </userAdminArray>
      </com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.35.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteHostingNEAdministrators"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.36 getHostingNEAdministrators

### 5.36.1 Description

This command allows a service provider to retrieve the list of administrator login IDs for a hosting network element.

### 5.36.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getHostingNEAdministrators"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0
        hostingNE="PSecho">
      </com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.36.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getHostingNEAdministrators"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0
        hostingNE="PSecho">
        <userAdminArray>
          <com.broadsoft.protocols.nsoss.UserAdminData
            userId="jsmith@telco.net"
            groupId="winnipeg">
          </com.broadsoft.protocols.nsoss.UserAdminData>
          <com.broadsoft.protocols.nsoss.UserAdminData
            userId="ppeterson@telco.net"
            enterprise="PrairiesEnt">
          </com.broadsoft.protocols.nsoss.UserAdminData>
        </userAdminArray>
      </com.broadsoft.protocols.nsoss.HostingNeAdminData_v19_0>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.37 addEnterpriseAdmin

### 5.37.1 Description

This command allows a service provider to create a new enterprise administrator in the system.

**NOTE 1:** The “userid” is required in the request and it must be unique across all levels of administrators in the system.

**NOTE 2:** The “password” is required in the request and it is transmitted in plain text.

**NOTE 3:** If the “firstName” attribute is not specified in the request, the Network Server assumes that the new enterprise administrator has a “blank” first name.

**NOTE 4:** If the “lastName” attribute is not specified in the request, the Network Server assumes that the new enterprise administrator has a “blank” last name.

**NOTE 5:** Only one enterprise administrator can be added per addEnterpriseAdmin command element. To add additional enterprise administrators, use additional addEnterpriseAdmin command elements.

### 5.37.2 Sample Input XML Structure

The following example assumes that enterprise “EnterpriseX” is provisioned as a valid enterprise in the system.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addEnterpriseAdmin"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseAdminData
        enterprise="EnterpriseX"
        firstName="John"
        lastName="Smith"
        password="myPassword"
        userid="jsmith">
      </com.broadsoft.protocols.nsoss.EnterpriseAdminData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.37.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addEnterpriseAdmin"
    loginId="admin"
```



```
result="0"
transactionId="1">
  <commandDataArray>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.38 deleteEnterpriseAdmin

### 5.38.1 Description

This command allows a service provider to delete an existing enterprise administrator from the system. The user ID must be defined in the system.

**NOTE 1:** All attributes other than "userid" specified in the request are ignored.

**NOTE 2:** Only one enterprise administrator can be deleted per deleteEnterpriseAdmin command element. To delete additional enterprise administrators, use additional deleteEnterpriseAdmin command elements.

### 5.38.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteEnterpriseAdmin"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseAdminData
        enterprise="EnterpriseX"
        userid="jsmith">
      </com.broadsoft.protocols.nsooss.EnterpriseAdminData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.38.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteEnterpriseAdmin"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.39 modifyEnterpriseAdmin

### 5.39.1 Description

This command allows a service provider to modify the properties of an existing enterprise administrator defined in the system.

**NOTE 1:** If the "firstName" attribute is not specified in the request, the Network Server assumes that the new first name of the enterprise administrator is "blank".

**NOTE 2:** If the "lastName" attribute is not specified in the request, the Network Server assumes that the new last name of the enterprise administrator is "blank".

**NOTE 3:** The "enterprise" information is required in the request but it is ignored by the Network Server. This means that the enterprise of an enterprise administrator cannot be changed after user ID has been created.

**NOTE 4:** If the "password" attribute is not specified in the request, the Network Server keeps the current password for the enterprise administrator. Otherwise, the new one is assigned to the enterprise administrator. Again, the password is transmitted in plain text.

**NOTE 5:** Only one enterprise administrator can be modified per modifyEnterpriseAdmin command element. To modify additional enterprise administrators, use additional modifyEnterpriseAdmin command elements.

### 5.39.2 Sample Input XML Structure

The following example assumes that user ID "jsmith" is provisioned as a valid enterprise administrator in the system.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyEnterpriseAdmin"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseAdminData
        enterprise="EnterpriseX"
        firstName="John A."
        lastName="Smith"
        password="myNewPassword"
        userid="jsmith">
      </com.broadsoft.protocols.nsoss.EnterpriseAdminData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.39.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
```

```
<com.broadsoft.protocols.nsooss.Command
  commandType="modifyEnterpriseAdmin"
  loginId="admin"
  result="0"
  transactionId="1">
  <commandDataArray>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.40 getEnterpriseAdmins

### 5.40.1 Description

This command allows a service provider to retrieve the properties of all existing enterprise administrators defined in the system.

**NOTE:** The "password" attribute is returned but it simply contains an empty string to hide the password content.

### 5.40.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getEnterpriseAdmins"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.40.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getEnterpriseAdmins"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseAdminData
        enterprise="EnterpriseX"
        firstName="John A."
        lastName="Smith"
        password=""
        userid="jsmith">
      </com.broadsoft.protocols.nsooss.EnterpriseAdminData>
    </com.broadsoft.protocols.nsooss.EnterpriseAdminData>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

```
        enterprise="EnterpriseY"  
        firstName="John"  
        lastName="Doe"  
        password=""  
        userid="jdoe">  
    </com.broadsoft.protocols.nsoss.EnterpriseAdminData>  
</commandDataArray>  
</com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.41 getEnterpriseAdmin

### 5.41.1 Description

This command allows a service provider to retrieve the properties of an existing enterprise administrator defined in the system. The user ID and enterprise name are required in the request and all other attributes are ignored when specified in the input XML string.

**NOTE 1:** The “enterprise” specified in the request is ignored because user IDs are unique in the system.

**NOTE 2:** The “password” attribute is returned but it simply contains an empty string to hide the password content.

**NOTE 3:** Only one enterprise administrator can be retrieved per getEnterpriseAdmin command element. To retrieve additional enterprise administrators, use additional getEnterpriseAdmin command elements.

### 5.41.2 Sample Input XML Structure

The following example assumes that user ID “jsmith” is provisioned as a valid enterprise administrator in the system.

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="getEnterpriseAdmin"  
    loginId="admin"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsoss.EnterpriseAdminData  
        enterprise="EnterpriseX"  
        userid="jsmith">  
      </com.broadsoft.protocols.nsoss.EnterpriseAdminData>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.41.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getEnterpriseAdmin"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseAdminData
        enterprise="EnterpriseX"
        firstName="John A."
        lastName="Smith"
        password=""
        userid="jsmith">
      </com.broadsoft.protocols.nsoss.EnterpriseAdminData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.42 synchronizeUserGroup

### 5.42.1 Description

This command allows a service provider to create a user group associated with an existing enterprise and hosting NE. If the user group exists, the operation has no effect. If the carrier information is not provided, the carrier is set to "NILCAC". If the enterprise is not specified, the default synchronized enterprise associated with the specified hosting NE is used. If the profile is not specified, the Network Server assigns the "NIL\_PROFILE" to the new user group. If the group name is not specified, the Network Server sets the group name to use the format "hostingNE – groupID". If the group name is specified, it must be unique within the group's enterprise. The group name is used for display purposes in the Network Server CommPilot web interface (for example, when selecting a group from a drop-down list).

A user group can be associated with an existing enterprise or to NIL\_ENTERPRISE. Carrier information at the user-group level overrides that at the enterprise level. If "NILCAC" is specified at the user-group level, then the corresponding carrier information at the enterprise level is used.

During route processing, both the user group and enterprise profiles are considered to determine the list of public policy instances that should process a call. If a policy has an instance in both profiles, the instance found in the user group profile is the one chosen for processing.

**NOTE 1:** Only one user group can be synchronized per synchronizeUserGroup command element. To synchronize more than one user group, use additional synchronizeUserGroup command elements.

**NOTE 2:** Specifying the DN/URL element has no effect for this command. To (un)assign DNs or URLs, use the (un)assignAddresses command.

**NOTE 3:** If specified, the lcaid must refer to an existing lcaid in the system. The lcaid can also be set to an empty string.

**NOTE 4:** The sourceId is optional. If specified, it is used to associate the user group to a source ID, or an originating trunk group (OTG).

### 5.42.2 Sample Input XML Structure

The following example assumes that:

- Carrier “Bell” is provisioned as a valid carrier for intraLATA, interLATA, and international calls.
- Profile “Profile1” is provisioned as a valid public routing profile in the system.
- Hosting NE “AS1” exists in the system, user group “GrpEA1” exists for hosting NE AS1 (under enterprise “EnterpriseX”), and “Grp#0” does not exist for hosting NE AS1.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeUserGroup"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.UserGroupData
        enterprise="EnterpriseX"
        hostingNE="AS1"
        groupId="Grp#0"
        interLataCarrier="Bell"
        internationalCarrier="Bell"
        intraLataCarrier="Bell"
        profile="Profile1"
        lcaid="DFLT_LCA_ID"
        sourceId="12345">
      </com.broadsoft.protocols.nsoss.UserGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
  <com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeUserGroup"
    loginId="admin"
    transactionId="2">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.UserGroupData
        groupId="GrpEA1"
        hostingNE="AS1">
      </com.broadsoft.protocols.nsoss.UserGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.42.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="synchronizeUserGroup "
    loginId="admin"
```

```
result="0"
transactionId="1">
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
commandType="synchronizeUserGroup "
loginId="admin"
result="0"
transactionId="2">
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.43 assignAddresses

### 5.43.1 Description

This command allows a service provider to assign one or more directory numbers and/or URLs to an existing user group. At the same time, a site can be assigned to each DN, URL, and DN range.

**NOTE 1:** Both the `dNArray` element (with any number of DNs, one DN per `DnData` element) and the `dNRangeArray` element (with any number of DN pairs representing the lower and upper limits of the inclusive DN ranges, one pair per `DnRangeData` element) may be provided.

**NOTE 2:** The `assignAddresses` command does not report an error if the addresses exist in the system and are assigned to the same user group. An error is returned only if one or more addresses to be assigned already belong to a user group different from the one specified in the command.

**NOTE 3:** A DN should be formatted as a global E.164 number composed of a country code, a national destination code, and a station code. In North America, this is an 11-digit phone number with 1 as the first digit, followed by a 3-digit Numbering Plan Area (NPA) and a 7-digit local number (NXX-XXXX).

**NOTE 4:** If a site is specified for the DN, DN range, or URL, the site must currently exist in the enterprise to which the user group belongs. If no site is specified, the addresses are assigned by default to site `DFLT_SITE` of the enterprise of the user group.

### 5.43.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="assignAddresses"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.UserGroupData
        groupID="Grp#0"
        hostingNE="AS1">
      <dNRangeArray>
        <com.broadsoft.protocols.nsoss.DnRangeData
          fromDn="12404455550"
          toDn="+12404455555"
```

```
        site="siteA">
        </com.broadsoft.protocols.nsoss.DnRangeData>
    </dnRangeArray>
    </com.broadsoft.protocols.nsoss.UserGroupData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="assignAddresses"
    loginId="admin"
    transactionId="2">
    <commandDataArray>
        <com.broadsoft.protocols.nsoss.UserGroupData
            groupID="Grp#0"
            hostingNE="AS1">
            <uRLArray>
                <com.broadsoft.protocols.nsoss.UrlData
                    url="aaaaa@broadsoft.com"
                    site="siteA"
                    isLoginId="true">
                </com.broadsoft.protocols.nsoss.UrlData>
            </uRLArray>
        </com.broadsoft.protocols.nsoss.UserGroupData>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.43.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
    <com.broadsoft.protocols.nsoss.Command
        commandType="assignAddresses"
        loginId="admin"
        result="0"
        transactionId="1">
    </com.broadsoft.protocols.nsoss.Command>
    <com.broadsoft.protocols.nsoss.Command
        commandType="assignAddresses"
        loginId="admin"
        result="0"
        transactionId="2">
    </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```



## 5.44 modifyAddresses

### 5.44.1 Description

This command allows a service provider to change the attributes of an existing directory numbers (and ranges) and/or URLs. Currently, the only attribute that can be changed is the site of the address.

**NOTE 1:** Both the dNArray element (with any number of DNs, one DN per DnData element) and the dNRangeArray element (with any number of DN pairs representing the lower and upper limits of the inclusive DN ranges, one pair per DnRangeData element) can be provided.

**NOTE 2:** The specified addresses must exist in the system to use this command. An error is returned if one or more addresses to be modified belong to a user group different from the one specified in the command.

**NOTE 3:** A DN should be formatted as a global E.164 number composed of a country code, a national destination code, and a station code. In North America, this is an 11-digit phone number with 1 as the first digit, followed by a 3-digit NPA and a 7-digit local number (NXX-XXXX).

**NOTE 4:** If a site is specified for the DN, DN range, or URL, the site must exist in the enterprise to which the user group belongs. If no site is specified, the addresses are moved to site DFLT\_SITE of the enterprise of the user group.

### 5.44.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyAddresses"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.UserGroupData
        groupID="Grp#0"
        hostingNE="AS1">
        <dNRangeArray>
          <com.broadsoft.protocols.nsoss.DnRangeData
            fromDn="12404455550"
            toDn="1240445555"
            site="siteB">
          </com.broadsoft.protocols.nsoss.DnRangeData>
        </dNRangeArray>
      </com.broadsoft.protocols.nsoss.UserGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyAddresses"
    loginId="admin"
    transactionId="2">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.UserGroupData
        groupID="Grp#0"
        hostingNE="AS1">
        <uRLArray>
```

```
        <com.broadsoft.protocols.nsoss.UrlData
            url="aaaaa@broadsoft.com"
            site="siteB"
            isLoginId="true">
        </com.broadsoft.protocols.nsoss.UrlData>
    </uRLArray>
    <com.broadsoft.protocols.nsoss.UserGroupData>
</commanddataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.44.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
    <com.broadsoft.protocols.nsoss.Command
        commandType="modifyAddresses"
        loginId="admin"
        result="0"
        transactionId="1">
    </com.broadsoft.protocols.nsoss.Command>
    <com.broadsoft.protocols.nsoss.Command
        commandType="assignAddresses"
        loginId="admin"
        result="0"
        transactionId="2">
    </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.45 unassignAddresses

### 5.45.1 Description

This command allows a service provider to unassign one or more directory numbers and/or URLs from an existing user group.

**NOTE 1:** Both the `dNArray` element (with any number of DNs, one DN per `DnData` element) and the `dNRangeArray` element (with any number of DN pairs representing the lower and upper limits of the inclusive DN ranges, one pair per `DnRangeData` element) may be provided.

**NOTE 2:** The `unassignAddresses` command does not report an error if the addresses do not exist in the system. An error is returned only if the addresses to be unassigned belong to a user group different from the one specified in the command.

**NOTE 3:** A DN should be formatted as a global E.164 number composed of a country code, a national destination code, and a station code. In North America, this is an 11-digit phone number with 1 as the first digit, followed by a 3-digit NPA and a 7-digit local number (NXX-XXXX).

**NOTE 4:** If a site is specified for the DN, DN range, or URL, the site must currently exist in the enterprise in which the user group belongs. If no site is specified, the addresses to be deleted are assumed to exist in site DFLT\_SITE of the enterprise of the user group.

### 5.45.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="unassignAddresses"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.UserGroupData
        groupID="Grp#0"
        hostingNE="AS1">
        <dNRangeArray>
          <com.broadsoft.protocols.nsooss.DnRangeData
            fromDn="12404455550"
            toDn="+12404455552">
          </com.broadsoft.protocols.nsooss.DnRangeData>
        </dNRangeArray>
      </com.broadsoft.protocols.nsooss.UserGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
  <com.broadsoft.protocols.nsooss.Command
    commandType="unassignAddresses"
    loginId="admin"
    transactionId="2">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.UserGroupData
        groupID="Grp#0"
        hostingNE="AS1">
        <uRLArray>
          <com.broadsoft.protocols.nsooss.UrlData
            url="bbbbbb@broadsoft.com"
            isLoginId="true">
          </com.broadsoft.protocols.nsooss.UrlData>
        </uRLArray>
      </com.broadsoft.protocols.nsooss.UserGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.45.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="unassignAddresses"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
```

```
<com.broadsoft.protocols.nsoss.Command
  commandType="unassignAddresses"
  loginId="admin"
  result="0"
  transactionId="2">
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.46 modifyUserGroup

### 5.46.1 Description

This command allows a service provider to modify existing user groups. When modifying a user group, all associated DNs/URLs remain unchanged in the Network Server (see the following note regarding the handling of sites). Although the DTD cannot capture this rule, the enterprise, carrier, lcaid, and sourcelid information must be provided when modifying the user group properties.

**NOTE 1:** If present in the request, the following elements are ignored by the Network Server: dNArray, dNRangeArray, and uRLArray.

**NOTE 2:** If the groupName is not provided, the user group's Network Server CommPilot web interface display name is set to "hostingNE - groupID". If the group name is specified, it must be unique within the group's enterprise.

**NOTE 3:** The enterprise provided must be provisioned as a valid enterprise in the system. When the enterprise is changed, all the addresses are migrated to an equivalent site in the new enterprise. If an address belongs to a site that does not exist in the new enterprise, the address is then moved to the DFLT\_SITE of the new enterprise.

**NOTE 4:** If specified, the lcaid must refer to an existing lcaid in the system. If the lcaid is set to an empty string, the user group no longer has its own local calling area.

**NOTE 5:** If specified, the sourcelid is used to associate the user group to a source ID, or originating trunk group (OTG). If a sourcelid is no longer required, set the sourcelid to an empty string.

**NOTE 6:** Only one user group can be modified per modifyUserGroup command element. To modify more than one user group, use additional modifyUserGroup command elements.

### 5.46.2 Sample Input XML Structure

The following example assumes that:

- Carrier "VTL" is provisioned as a valid carrier for intraLATA, interLATA, and international calls.
- Profile "Profile2" is provisioned as a valid public routing profile in the system. If the profile is not specified in the request, the Network Server leaves the user group profile as is and does not change its current value.

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
```

```
<com.broadsoft.protocols.nsooss.Command
  commandType="modifyUserGroup"
  loginId="admin"
  transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsooss.UserGroupData
      enterprise="EnterpriseX"
      groupId="Grp#0"
      hostingNE="AS1"
      interLataCarrier="VTL"
      internationalCarrier="VTL"
      intraLataCarrier="VTL"
      profile="Profile2"
      lcaid="grp0_LCA"
      sourceId="4567">
    </com.broadsoft.protocols.nsooss.UserGroupData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.46.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyUserGroup"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.47 deleteUserGroup

### 5.47.1 Description

This command allows a service provider to delete user groups associated to hosting network elements.

**NOTE 1:** Only one user group can be deleted per deleteUserGroup command element. To delete more than one user group, use additional deleteUserGroup command elements.

**NOTE 2:** Directory numbers and URLs assigned to the user group are unassigned.

### 5.47.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteUserGroup"
    loginId="admin"
```

```
transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsooss.UserGroupData
      groupID="Grp#0"
      hostingNE="AS1">
    </com.broadsoft.protocols.nsooss.UserGroupData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.47.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteUserGroup"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.48 getUserGroup

### 5.48.1 Description

This command allows a service provider to query an existing user group in the Network Server. Information on all assigned DNS and URLs is returned as well.

### 5.48.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getUserGroup"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.UserGroupData
        groupID="Grp#0"
        hostingNE="AS1">
      </com.broadsoft.protocols.nsooss.UserGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.48.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
```

```
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getUserGroup"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.UserGroupData
        enterprise="EnterpriseX"
        groupID="Grp#0"
        groupName="AS1 - Grp#0"
        hostingNE="AS1"
        interLataCarrier="VTL"
        internationalCarrier="VTL"
        intraLataCarrier="VTL"
        profile="Profile2"
        lcaid="grp0_LCA"
        sourceId="4567">
      <dNArray>
        <com.broadsoft.protocols.nsooss.DnData
          dn="12404661234"
          site="DFLT_SITE">
        </com.broadsoft.protocols.nsooss.DnData>
        <com.broadsoft.protocols.nsooss.DnData
          dn="12404661240"
          site="siteA">
        </com.broadsoft.protocols.nsooss.DnData>
        <com.broadsoft.protocols.nsooss.DnData
          dn="12404661250"
          site="siteB">
        </com.broadsoft.protocols.nsooss.DnData>
      </dNArray>
      <dNRangeArray>
        <com.broadsoft.protocols.nsooss.DnRangeData
          fromDn="12404662000"
          toDn="12404662010"
          site="DFLT_SITE">
        </com.broadsoft.protocols.nsooss.DnRangeData>
        <com.broadsoft.protocols.nsooss.DnRangeData
          fromDn="12404662020"
          toDn="12404662030"
          site="siteB">
        </com.broadsoft.protocols.nsooss.DnRangeData>
      </dNRangeArray>
      <uRLArray>
        <com.broadsoft.protocols.nsooss.UrlData
          isLoginId="true"
          url="aaaaa@broadsoft.com"
          site="DFLT_SITE">
        </com.broadsoft.protocols.nsooss.UrlData>
        <com.broadsoft.protocols.nsooss.UrlData
          isLoginId="true"
          url="bbbbb@broadsoft.com"
          site="site2">
        </com.broadsoft.protocols.nsooss.UrlData>
      </uRLArray>
    </com.broadsoft.protocols.nsooss.UserGroupData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.49 getUserGroups

### 5.49.1 Description

This command allows a service provider to query all user groups on the Network Server. Information on all assigned DNS and URLs per user group is returned as well.

### 5.49.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getUserGroups"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getUserGroups"
    loginId="admin"
    transactionId="2">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.UserGroupData
        groupID="Grp#0"
        hostingNE="AS1">
      </com.broadsoft.protocols.nsooss.UserGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.49.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getUserGroups"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.UserGroupData
        enterprise="EnterpriseX"
        groupID="Grp#0"
        groupName="AS1 - Grp#0"
        hostingNE="AS1"
        interLataCarrier="VTL"
        internationalCarrier="VTL"
        intraLataCarrier="VTL"
        profile="Profile2"
        lcaid="grp0_LCA"
        sourceId="4567">
        <dNArray>
          <com.broadsoft.protocols.nsooss.DnData
            dn="12404661234"
            site="DFLT_SITE">
          </com.broadsoft.protocols.nsooss.DnData>
          <com.broadsoft.protocols.nsooss.DnData
```



```
        dn="12404661240"
        site="siteA">
</com.broadsoft.protocols.nsoss.DnData>
<com.broadsoft.protocols.nsoss.DnData
  dn="12404661250"
  site="siteB">
</com.broadsoft.protocols.nsoss.DnData>
</dnArray>
<dNRangeArray>
  <com.broadsoft.protocols.nsoss.DnRangeData
    fromDn="12404662000"
    toDn="12404662010"
    site="DFLT_SITE">
</com.broadsoft.protocols.nsoss.DnRangeData>
  <com.broadsoft.protocols.nsoss.DnRangeData
    fromDn="12404662020"
    toDn="12404662030"
    site="siteB">
</com.broadsoft.protocols.nsoss.DnRangeData>
</dNRangeArray>
<uRLArray>
  <com.broadsoft.protocols.nsoss.UrlData
    isLoginId="true"
    url="aaaaa@broadsoft.com"
    site="DFLT_SITE">
</com.broadsoft.protocols.nsoss.UrlData>
  <com.broadsoft.protocols.nsoss.UrlData
    isLoginId="true"
    url="bbbbbb@broadsoft.com"
    site="site2">
</com.broadsoft.protocols.nsoss.UrlData>
</uRLArray>
</com.broadsoft.protocols.nsoss.UserGroupData>
<com.broadsoft.protocols.nsoss.UserGroupData
  enterprise="EnterpriseX"
  groupID="GrpEA1"
  groupName="AS1 - GrpEA1"
  hostingNE="AS1"
  interLataCarrier="Bell"
  internationalCarrier="Bell"
  intraLataCarrier="Bell"
  profile="Profile1"
  lcaid="DFLT_LCA_ID"
  sourceId="">
<dnArray>
</dnArray>
<dNRangeArray>
</dNRangeArray>
<uRLArray>
</uRLArray>
</com.broadsoft.protocols.nsoss.UserGroupData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
  commandType="getUserGroups"
  loginId="admin"
  result="0"
  transactionId="2">
  <commandDataArray>
    <com.broadsoft.protocols.nsoss.UserGroupData
      enterprise="EnterpriseX"
      groupID="Grp#0"
      groupName="AS1 - Grp#0"
```

```
hostingNE="AS1"
interLataCarrier="VTL"
internationalCarrier="VTL"
intraLataCarrier="VTL"
profile="Profile2"
lcaid="grp0_LCA"
sourceId="4567">
<dNArray>
  <com.broadsoft.protocols.nsoss.DnData
    dn="12404661234"
    site="DFLT_SITE">
  </com.broadsoft.protocols.nsoss.DnData>
  <com.broadsoft.protocols.nsoss.DnData
    dn="12404661240"
    site="siteA">
  </com.broadsoft.protocols.nsoss.DnData>
  <com.broadsoft.protocols.nsoss.DnData
    dn="12404661250"
    site="siteB">
  </com.broadsoft.protocols.nsoss.DnData>
</dNArray>
<dNRangeArray>
  <com.broadsoft.protocols.nsoss.DnRangeData
    fromDn="12404662000"
    toDn="12404662010"
    site="DFLT_SITE">
  </com.broadsoft.protocols.nsoss.DnRangeData>
  <com.broadsoft.protocols.nsoss.DnRangeData
    fromDn="12404662020"
    toDn="12404662030"
    site="siteB">
  </com.broadsoft.protocols.nsoss.DnRangeData>
</dNRangeArray>
<uRLArray>
  <com.broadsoft.protocols.nsoss.UrlData
    isLoginId="true"
    url="aaaaa@broadsoft.com"
    site="DFLT_SITE">
  </com.broadsoft.protocols.nsoss.UrlData>
  <com.broadsoft.protocols.nsoss.UrlData
    isLoginId="true"
    url="bbbbbb@broadsoft.com"
    site="site2">
  </com.broadsoft.protocols.nsoss.UrlData>
</uRLArray>
  </com.broadsoft.protocols.nsoss.UserGroupData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.50 addAuthorizedHost

### 5.50.1 Description

This command allows a service provider to authorize a FarEndHopOff policy instance or an EntNGWRouting policy instance to use hosting NEs provisioned on the Network Server for routing. A FarEndHopOff or EntNGWRouting policy instance can only route calls to a hosting NE if this hosting NE is first authorized to be used by the instance. To use this command, both the enterprise and hosting NE must exist on the Network Server.

**NOTE:** Only one authorized host can be added per addAuthorizedHost command element. To add additional authorized hosts, use additional addAuthorizedHost command elements.

### 5.50.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addAuthorizedHost"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.AuthorizedHostData
        enterprise="EnterpriseX"
        policy="FarEndHopOff"
        authHost="AS1">
      </com.broadsoft.protocols.nsoss.AuthorizedHostData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.50.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addAuthorizedHost"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.51 deleteAuthorizedHost

### 5.51.1 Description

This command allows a service provider to un-authorize a FarEndHopOff or EntNGWRouting policy instance that is using a hosting NE provisioned on the Network Server for routing. A FarEndHopOff or EntNGWRouting policy instance can only route calls to a hosting NE if this hosting NE is authorized to be used by the instance. To use this command, both the enterprise and hosting NE must exist on the Network Server.

**NOTE:** Only one authorized host can be deleted per deleteAuthorizedHost command element. To delete additional authorized hosts, use additional deleteAuthorizedHost command elements.

### 5.51.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteAuthorizedHost"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.AuthorizedHostData
        enterprise="EnterpriseX"
        policy="FarEndHopOff"
        authHost="AS1">
      </com.broadsoft.protocols.nsoss.AuthorizedHostData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.51.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteAuthorizedHost"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.52 getAuthorizedHosts

### 5.52.1 Description

This command allows a service provider to retrieve the list of all the authorized hosts for an instance of the FarEndHopOff or EntNGWRouting private policies.

### 5.52.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
commandType="getAuthorizedHosts"
  loginId="admin"
  transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.AuthorizedHostData
enterprise="EnterpriseX"
policy="FarEndHopOff">
      </com.broadsoft.protocols.nsooss.AuthorizedHostData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.52.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
commandType="getAuthorizedHosts"
  loginId="admin"
  result="0"
  transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.AuthorizedHostData
enterprise="EnterpriseX"
policy="FarEndHopOff"
authHost="AS1">
      </com.broadsoft.protocols.nsooss.AuthorizedHostData>
      <com.broadsoft.protocols.nsooss.AuthorizedHostData
enterprise="EnterpriseX"
policy="FarEndHopOff"
authHost="AS2">
      </com.broadsoft.protocols.nsooss.AuthorizedHostData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.53 getCallTypes

### 5.53.1 Description

This command allows a system provider to query the Network Server for all existing call types.

### 5.53.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getCallTypes"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.53.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getCallTypes"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.CallTypeData
        callType="LO">
      </com.broadsoft.protocols.nsoss.CallTypeData>
      <com.broadsoft.protocols.nsoss.CallTypeData
        callType="TO">
      </com.broadsoft.protocols.nsoss.CallTypeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.54 getTreatments

### 5.54.1 Description

This command allows a system provider to query the Network Server for all existing treatments.

### 5.54.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getTreatments"
```

```
loginId="admin"  
transactionId="1">  
</com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.54.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="getTreatments"  
    loginId="admin"  
    result="0"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsoss.TreatmentData  
        treatmentID="invdd">  
      </com.broadsoft.protocols.nsoss.TreatmentData>  
      <com.broadsoft.protocols.nsoss.TreatmentData  
        treatmeID="undge">  
      </com.broadsoft.protocols.nsoss.TreatmentData>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.55 addPrivatePolicyInstanceDataEntry

### 5.55.1 Description

This command allows a system provider to add enterprise policy instance data entries to the system. Only the EntNGWRouting and voiceVPN policies can be configured using the OSS interface.

**NOTE 1:** Only one private policy instance data entry can be added per addPrivatePolicyInstanceDataEntry command element. To add additional private policy instance data entries, use additional addPrivatePolicyInstanceDataEntry command elements.

**NOTE 2:** Attribute "enterprise" must refer to an existing enterprise in the system. It is required in EntNGWRoutingRoutingData and VoiceVPNRoutingData.

**NOTE 3:** Attribute "site" must refer to an existing site in the enterprise specified in the request. It is required in EntNGWRoutingRoutingData.

**NOTE 4:** Attribute "callType" must refer to an existing call type in the system. It is required in EntNGWRoutingRoutingData. The call type used in EntNGWRoutingRoutingData must first be authorized using the EntNGWRoutingPropertiesData element.

**NOTE 5:** Attribute "authHost" must refer to an existing hosting NE flagged as being an authorized host for the enterprise policy. It is required in EntNGWRoutingRoutingData.

**NOTE 6:** Attribute “cost” is an integer between 1 and 9. During route selection, the Network Server returns possible routes in cost ascending order. This attribute is required in EntNGWRoutingRoutingData.

**NOTE 7:** Attribute “weight” is an integer between 0 and 99. During route selection, the weight is used to sort routes of equal cost. This attribute is required in EntNGWRoutingRoutingData.

**NOTE 8:** Attribute “outDM” is a string that can have 0 to 80 characters. It uses the standard Network Server digit manipulation syntax (see [Appendix A: Network Server Digit Manipulation](#)) to describe how the resulting number should be altered when added as a routing result. This attribute is required in EntNGWRoutingRoutingData. In VoiceVPNRoutingData, “outDM” is optional and cannot be used if the corresponding selector is set to “TRMT”.

**NOTE 9:** Attribute “description” is a string that can have 0 up to 40 characters. This attribute is optional in VoiceVPNRoutingData.

**NOTE 10:** Attribute “locCode” is a string that can have 1 up to 15 characters. It identifies the location code of a user group in an enterprise. This attribute is required in VoiceVPNRoutingData.

**NOTE 11:** Attributes “extMin” and “extMax” are integers from the range (0 to 100). They specify the format of an extension within a user group. They are required in VoiceVPNRoutingData.

**NOTE 12:** Attribute “selector” can be one of the following values: “PRIVATE”, “PUBLIC”, “ROUTE”, or “TRMT”. It identifies the routing behavior of an entry. This attribute is required in VoiceVPNRoutingData.

**NOTE 13:** Attribute “hostingNE” must refer to an existing hosting NE in the system. It is mandatory in VoiceVPNRoutingData if the corresponding selector is “ROUTE”, and disallowed in all other cases.

**NOTE 14:** Attribute “groupID” must refer to an existing user group hosted on the hosting NE also specified. It is mandatory in VoiceVPNRoutingData if the corresponding selector is “ROUTE”, and disallowed in all other cases.

**NOTE 15:** Attribute “treatmentID” must refer to an existing treatment in the system. It is required in VoiceVPNRoutingData (only if selector “TRMT” is used; otherwise it is disallowed). The treatment used in VoiceVPNRoutingData must first be authorized using the VoiceVPNPropertiesData element.

### 5.55.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addPrivatePolicyInstanceDataEntry"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingD
ata
  enterprise="EnterpriseX"
```



```

        site="siteA"
        callType="LO"
        authHost="Host1"
        cost="2"
        weight="50"
        outDM="POS (0) , DEL (1) ">
</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRouting
Data>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="addPrivatePolicyInstanceDataEntry"
    loginId="admin"
    transactionId="2">
    <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
    enterprise="EnterpriseX"
    locCode="7"
    extMin="4"
    extMax="4"
    selector="ROUTE"
    outDM="POS (0) , DEL (1) "
    hostingNE="Host1"
    groupID="Group1"
    description="Host1-Group1 VPN entry with location code 7">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingDat
a>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

### 5.55.3 Sample Output XML Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
    <com.broadsoft.protocols.nsoss.Command
        commandType="addPrivatePolicyInstanceDataEntry"
        loginId="admin"
        result="0"
        transactionId="1">
    </com.broadsoft.protocols.nsoss.Command>
    <com.broadsoft.protocols.nsoss.Command
        commandType="addPrivatePolicyInstanceDataEntry"
        loginId="admin"
        result="0"
        transactionId="2">
    </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

## 5.56 addPrivatePolicyPropertiesDataEntry

### 5.56.1 Description

This command allows a system provider to add enterprise policy data entries to the system. Only the EntNGWRouting and voiceVPN policies can be configured using the OSS interface.

**NOTE 1:** Only one private policy properties data entry can be added per addPrivatePolicyPropertiesDataEntry command element. To add additional private policy properties data entries, use additional addPrivatePolicyPropertiesDataEntry command elements.

**NOTE 2:** Attribute "callType" must refer to an existing call type in the system. It is required in EntNGWRoutingPropertiesData.

**NOTE 3:** Attribute "description" is a string that can have 1 to 40 characters (30 in VoiceVPNPropertiesData) characters. This attribute is required in EntNGWRoutingPropertiesData and VoiceVPNPropertiesData.

**NOTE 4:** Attribute "treatmentID" must refer to an existing treatment in the system. It is required in VoiceVPNPropertiesData.

### 5.56.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="1">
    <commandDataArray>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData
      callType="DA"
      description="Directory Assistance for EntNGWRouting">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="2">
    <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesD
ata
      treatmentID="bdreq"
      description="SIP format Error">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertie
sData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
```

```
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.56.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
  version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="addPrivatePolicyPropertiesDataEntry"  
    loginId="admin"  
    result="0"  
    transactionId="1">  
  </com.broadsoft.protocols.nsooss.Command>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="addPrivatePolicyPropertiesDataEntry"  
    loginId="admin"  
    result="0"  
    transactionId="2">  
  </com.broadsoft.protocols.nsooss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.57 modifyPrivatePolicyInstanceDataEntry

### 5.57.1 Description

This command allows a system provider to modify enterprise policy instance data entries defined in the system. Only the EntNGWRouting and voiceVPN policies can be re-configured using the OSS interface.

**NOTE 1:** Only one private policy instance data entry can be modified per modifyPrivatePolicyInstanceDataEntry command element. To modify additional private policy instance data entries, use additional modifyPrivatePolicyInstanceDataEntry command elements.

**NOTE 2:** Attribute "enterprise" must refer to an existing enterprise in the system. It is required in EntNGWRoutingRoutingData and VoiceVPNRoutingData.

**NOTE 3:** Attribute "site" must refer to an existing site in the enterprise specified in the request. It is required in EntNGWRoutingRoutingData.

**NOTE 4:** Attribute "callType" must refer to an existing call type in the system. It is required in EntNGWRoutingRoutingData. The call type used in EntNGWRoutingRoutingData must first be authorized using the EntNGWRoutingPropertiesData element.

**NOTE 5:** Attribute "authHost" must refer to an existing hosting NE flagged as being an authorized host for the enterprise policy. It is required in EntNGWRoutingRoutingData.

**NOTE 6:** Attribute "cost" is an integer between 1 and 9. During route selection, the Network Server returns possible routes in cost ascending order. This attribute is optional in EntNGWRoutingRoutingData.

**NOTE 7:** Attribute "weight" is an integer between 0 and 99. During route selection, the weight is used to sort routes of equal cost. This attribute is optional in EntNGWRoutingRoutingData.

**NOTE 8:** Attribute “outDM” is a string that can have 0 to 80 characters. It uses the standard Network Server digit manipulation syntax (for more information, see [Appendix A: Network Server Digit Manipulation](#)) to describe how the resulting number should be altered when added as a routing result. This attribute is optional in EntNGWRoutingRoutingData. In VoiceVPNRoutingData, “outDM” is optional and cannot be used if the corresponding selector is set to “TRMT”.

**NOTE 9:** Attribute “description” is a string that can have 0 to 40 characters. This attribute is optional in VoiceVPNRoutingData.

**NOTE 10:** Attribute “locCode” is a string that can have 1 to 15 characters. It identifies the location code of a user group in an enterprise. This attribute is required in VoiceVPNRoutingData.

**NOTE 11:** Attributes “extMin” and “extMax” are integers in range (from 0 to 100). They specify the format of an extension within a user group. They are optional in VoiceVPNRoutingData.

**NOTE 12:** Attribute “selector” can take one of the following values: “PRIVATE”, “PUBLIC”, “ROUTE”, or “TRMT”. It identifies the routing behavior of an entry. This attribute is optional in VoiceVPNRoutingData.

**NOTE 13:** Attribute “hostingNE” must refer to an existing hosting NE in the system. It is mandatory in VoiceVPNRoutingData if the corresponding selector is changed to “ROUTE”, optional if the selector is “ROUTE”, and disallowed in all other cases.

**NOTE 14:** Attribute “groupID” needs to refer to an existing user group. It is mandatory in VoiceVPNRoutingData if the corresponding selector is changed to “ROUTE”, optional if the selector is “ROUTE”, and disallowed in all other cases.

**NOTE 15:** Attribute “treatmentID” must refer to an existing treatment in the system. It is required in VoiceVPNRoutingData (only if the selector is changed to “TRMT”; otherwise it is disallowed or optional if “TRMT” is already set). The treatment used in VoiceVPNRoutingData must first be authorized using the VoiceVPNPropertiesData element.

### 5.57.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyPrivatePolicyInstanceDataEntry"
    loginId="admin"
    transactionId="1">
    <commandDataArray>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingD
ata
  enterprise="EnterpriseX"
  site="siteA"
  callType="LO"
  authHost="Host1"
  cost="6"
  weight="75"
  outDM="POS (0) , DEL (2) ">
```

```

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRouting
Data>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="modifyPrivatePolicyInstanceDataEntry"
    loginId="admin"
    transactionId="2">
    <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
    enterprise="EnterpriseX"
    locCode="7"
    extMin="5"
    extMax="5"
    selector="ROUTE"
    outDM="POS (0) , DEL (2) "
    hostingNE="Host2"
    groupID="Group2"
    description="Host2-Group2 VPN entry with location code 7">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingDat
a>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

### 5.57.3 Sample Output XML Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
    <com.broadsoft.protocols.nsoss.Command
        commandType="modifyPrivatePolicyInstanceDataEntry"
        loginId="admin"
        result="0"
        transactionId="1">
    </com.broadsoft.protocols.nsoss.Command>
    <com.broadsoft.protocols.nsoss.Command
        commandType="modifyPrivatePolicyInstanceDataEntry"
        loginId="admin"
        result="0"
        transactionId="2">
    </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

## 5.58 modifyPrivatePolicyPropertiesDataEntry

### 5.58.1 Description

This command allows a system provider to modify enterprise policy data entries defined in the system. Only policies EntNGWRouting and voiceVPN can be re-configured using the OSS interface.

**NOTE 1:** Only one private policy properties data entry can be modified per modifyPrivatePolicyPropertiesDataEntry command element. To modify additional private policy properties data entries, use additional modifyPrivatePolicyPropertiesDataEntry command elements.

**NOTE 2:** Attribute "callType" needs to refer to an existing call type in the system. It is required in EntNGWRoutingPropertiesData.

**NOTE 3:** Attribute "description" is a string that can have one to 40 characters (30 characters in VoiceVPNPropertiesData). This attribute is required in EntNGWRoutingPropertiesData and VoiceVPNPropertiesData.

**NOTE 4:** Attribute "treatmentID" must refer to an existing treatment in the system. It is required in VoiceVPNPropertiesData.

### 5.58.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="1">
    <commandDataArray>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData
      callType="DA"
      description="Directory Assistance for policy EntNGWRouting">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="2">
    <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesD
ata
      treatmentID="bdreq"
      description="SIP Format Error">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNProperties
Data>
    </commandDataArray>
```

```
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.58.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
  version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    result="0"
    transactionId="2">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.59 getPrivatePolicyInstanceDataEntry

### 5.59.1 Description

This command allows a system provider to retrieve enterprise policy instance data entries defined in the system. Only policies EntNGWRouting and voiceVPN can be queried using the OSS interface.

**NOTE 1:** Only data entries of one private policy instance can be retrieved per `getPrivatePolicyInstanceDataEntry` command element. To retrieve data for additional private policy instances, use additional `getPrivatePolicyInstanceDataEntry` command elements.

**NOTE 2:** Attribute "enterprise" must refer to an existing enterprise in the system. It is required in `EntNGWRoutingRoutingData` and `VoiceVPNRoutingData`.

**NOTE 3:** This command returns all the enterprise policy instance rows if no other attribute is specified. It is also possible to retrieve specific rows by providing the key attributes as input, namely "site", "callType", and "authHost" for `EntNGWRouting`, or "locCode" for `voiceVPN`.

### 5.59.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
  version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getPrivatePolicyInstanceDataEntry"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
```

```

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingD
ata
    enterprise="EnterpriseX">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRouting
Data>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="getPrivatePolicyInstanceDataEntry"
    transactionId="2">
    loginId="admin"
    <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
    enterprise="EnterpriseX">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingDat
a>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

### 5.59.3 Sample Output XML Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
    version="20.0">
<commandArray>
    <com.broadsoft.protocols.nsoss.Command
        commandType="getPrivatePolicyInstanceDataEntry"
        loginId="admin"
        result="0"
        transactionId="1">
        <commandDataArray>
<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingD
ata
    authHost="Host1"
    callType="LO"
    cost="6"
    enterprise="EnterpriseX"
    outDM="POS (0), DEL (2) "
    site="siteA"
    weight="75">
</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRouting
Data>
<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingD
ata
    authHost="Host2"
    callType="TO"
    cost="1"
    enterprise="EnterpriseX"
    outDM=""
    site="siteA"
    weight="1">
</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRouting
Data>
<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingD
ata

```



```
        authHost="Host1"
        callType="ALL"
        cost="2"
        enterprise="EnterpriseX"
        outDM=""
        site="siteA"
        weight="99">
</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRouting
Data>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
    commandType="getPrivatePolicyInstanceDataEntry"
    loginId="admin"
    result="0"
    transactionId="2">
    <commandDataArray>
<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
description="Host2-Group2 VPN entry with location code 7"
enterprise="EnterpriseX"
extMax="5"
extMin="5"
groupID="Group2"
hostingNE="Host2"
    locCode="7"
    outDM="POS (0) , DEL (2) "
    selector="ROUTE">
</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingDat
a>
<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
description="location code 4"
enterprise="EnterpriseX"
extMax="4"
extMin="4"
    locCode="4"
    outDM=""
    selector="PUBLIC">
</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingDat
a>
<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
description="treatment entry"
enterprise="EnterpriseX"
extMax="7"
extMin="7"
    locCode="2"
    selector="TRMT"
treatmentID="abcde">
</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingDat
a>
    </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.60 getPrivatePolicyPropertiesDataEntry

### 5.60.1 Description

This command allows a system provider to retrieve enterprise policy data entries defined in the system. Only policies EntNGWRouting and voiceVPN can be queried using the OSS interface.

**NOTE 1:** Only data entries of one private policy can be retrieved per getPrivatePolicyPropertiesDataEntry command element. To retrieve data for additional private policies, use additional getPrivatePolicyPropertiesDataEntry command elements.

**NOTE 2:** This command returns all the enterprise policy rows if no other attribute is specified. It is also possible to retrieve specific rows, by providing the key attributes as input, namely "callType" for EntNGWRouting, or "treatmentID" for voiceVPN.

### 5.60.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="1">
    <commandDataArray>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData>

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getPrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="2">
    <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesD
ata>

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNProperties
Data>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.60.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
```

```
commandType="getPrivatePolicyPropertiesDataEntry"
loginId="admin"
result="0"
transactionId="1">
  <commandDataArray>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData
  callType="DA"
  description="Directory Assistance for policy EntNGWRouting">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData
  callType="LO"
  description="Local">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData
  callType="TO"
  description="Toll">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData
  callType="ALL"
  description="All call types">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>
  </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
<com.broadsoft.protocols.nsoss.Command
  commandType="getPrivatePolicyPropertiesDataEntry"
  result="0"
  loginId="admin"
  transactionId="2">
  <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesD
ata
description="SIP Format Error"
treatmentID="bdreq">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNProperties
Data>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesD
ata
description="Custom treatment"
treatmentID="abcde">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNProperties
Data>
  </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
```

```
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.61 deletePrivatePolicyInstanceDataEntry

### 5.61.1 Description

This command allows a system provider to delete enterprise policy instance data entries defined in the system. Only EntNGWRouting and voiceVPN data entries can be deleted using the OSS interface.

**NOTE 1:** Only one private policy instance data entry can be deleted per deletePrivatePolicyInstanceDataEntry command element. To delete additional private policy instance data entries, use additional deletePrivatePolicyInstanceDataEntry command elements.

**NOTE 2:** Attribute "enterprise" must refer to an existing enterprise in the system. It is required in EntNGWRoutingRoutingData and VoiceVPNRoutingData.

**NOTE 3:** Attribute "site" must refer to an existing site in the enterprise specified in the request. It is required in EntNGWRoutingRoutingData.

**NOTE 4:** Attribute "callType" must refer to an existing call type in the system. It is required in EntNGWRoutingRoutingData. The call type used in EntNGWRoutingRoutingData needs to first be authorized using the EntNGWRoutingPropertiesData element.

**NOTE 5:** Attribute "authHost" must refer to an existing hosting NE flagged as being an authorized host for the enterprise policy. It is required in EntNGWRoutingRoutingData.

**NOTE 6:** Attribute "locCode" is a string that can have 1 to 15 characters. It identifies the location code of a user group in an enterprise. This attribute is required in VoiceVPNRoutingData.

### 5.61.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="deletePrivatePolicyInstanceDataEntry"  
    loginId="admin"  
    transactionId="1">  
    <commandDataArray>  
  
<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRoutingD  
ata  
    enterprise="EnterpriseX"  
    site="siteA"  
    callType="LO"  
    authHost="Host1">  
  
</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingRouting  
Data>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</com.broadsoft.protocols.nsoss.Command
```

```

commandType="deletePrivatePolicyInstanceDataEntry"
loginId="admin"
transactionId="2">
  <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingData
  enterprise="EnterpriseX"
  locCode="7">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNRoutingDat
a>
  </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

### 5.61.3 Sample Output XML Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
  version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deletePrivatePolicyInstanceDataEntry"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deletePrivatePolicyInstanceDataEntry"
    loginId="admin"
    result="0"
    transactionId="2">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>

```

## 5.62 deletePrivatePolicyPropertiesDataEntry

### 5.62.1 Description

This command allows a system provider to delete enterprise policy data entries defined in the system. Only EntNGWRouting and voiceVPN data entries can be deleted using the OSS interface.

**NOTE 1:** Only one private policy properties data entry can be deleted per deletePrivatePolicyPropertiesDataEntry command element. To delete additional private policy properties data entries, use additional deletePrivatePolicyPropertiesDataEntry command elements.

**NOTE 2:** Attribute "callType" must refer to an existing call type in the system. It is required in EntNGWRoutingPropertiesData.

**NOTE 3:** Attribute "treatmentID" must refer to an existing treatment in the system. It is required in VoiceVPNPropertiesData.

## 5.62.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deletePrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="1">
    <commandDataArray>

<com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingProperti
esData
    callType="DA">

</com.broadsoft.routingserver.policies.entngwrouting.EntNGWRoutingPropert
iesData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deletePrivatePolicyPropertiesDataEntry"
    loginId="admin"
    transactionId="2">
    <commandDataArray>

<com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNPropertiesD
ata
    treatmentID="bdreq">

</com.broadsoft.routingserver.policies.privatedialplan.VoiceVPNProperties
Data>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.62.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deletePrivatePolicyPropertiesDataEntry"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deletePrivatePolicyPropertiesDataEntry"
    loginId="admin"
    result="0"
    transactionId="2">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.63 addUnassignedNumber

### 5.63.1 Description

This command allows a system provider to add one or more unassigned directory numbers to the system.

**NOTE 1:** Both the `dNArray` element (with any number of DNs, one DN per `DnData` element) and the `dNRangeArray` element (with any number of DN pairs representing the lower and upper limits of the inclusive DN ranges, one pair per `DnRangeData` element) may be provided. The `site` attribute is ignored.

**NOTE 2:** A DN should be formatted as a global E.164 number composed of a country code, a national destination code, and a station code. In North America, this is an 11-digit phone number with 1 as the first digit, followed by a 3-digit NPA, and a 7-digit local number (NXX-XXXX).

### 5.63.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addUnassignedNumber"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.UnassignedNumberData>
        <dNRangeArray>
          <com.broadsoft.protocols.nsoss.DnRangeData
            fromDn="12404455550"
            toDn="+1240445555">
          </com.broadsoft.protocols.nsoss.DnRangeData>
          <com.broadsoft.protocols.nsoss.DnRangeData
            fromDn="12404457777"
            toDn="+12404457779">
          </com.broadsoft.protocols.nsoss.DnRangeData>
        </dNRangeArray>
        <dNArray>
          <com.broadsoft.protocols.nsoss.DnData
            dn="12404451111">
          </com.broadsoft.protocols.nsoss.DnData>
          <com.broadsoft.protocols.nsoss.DnData
            dn="12404450000">
          </com.broadsoft.protocols.nsoss.DnData>
        </dNArray>
      </com.broadsoft.protocols.nsoss.UnassignedNumberData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.63.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
```

```
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addUnassignedNumber"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.64 deleteUnassignedNumber

### 5.64.1 Description

This command allows a system provider to delete one or more unassigned directory numbers from the system.

**NOTE 1:** Both the `dNArray` element (with any number of DNs, one DN per `DnData` element) and the `dNRangeArray` element (with any number of DN pairs representing the lower and upper limits of the inclusive DN ranges, one pair per `DnRangeData` element) can be provided. The site attribute is ignored.

**NOTE 2:** A DN should be formatted as a global E.164 number composed of a country code, a national destination code, and a station code. In North America, this is an 11-digit phone number with 1 as the first digit, followed by a 3-digit NPA, and a 7-digit local number (NXX-XXXX).

### 5.64.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteUnassignedNumber"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.UnassignedNumberData>
        <dNRangeArray>
          <com.broadsoft.protocols.nsooss.DnRangeData
            fromDn="12404455550"
            toDn="+12404455555">
          </com.broadsoft.protocols.nsooss.DnRangeData>
          <com.broadsoft.protocols.nsooss.DnRangeData
            fromDn="12404457777"
            toDn="+12404457779">
          </com.broadsoft.protocols.nsooss.DnRangeData>
        </dNRangeArray>
        <dNArray>
          <com.broadsoft.protocols.nsooss.DnData
            dn="12404451111">
          </com.broadsoft.protocols.nsooss.DnData>
          <com.broadsoft.protocols.nsooss.DnData
            dn="12404450000">
          </com.broadsoft.protocols.nsooss.DnData>
        </dNArray>
      </com.broadsoft.protocols.nsooss.UnassignedNumberData>
```



```
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.64.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteUnassignedNumber"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.65 getUnassignedNumber

### 5.65.1 Description

This command allows a system provider to query all unassigned directory numbers defined in the system.

**NOTE 1:** Both the `dNArray` element (with any number of DNs, one DN per `DnData` element) and the `dNRangeArray` element (with any number of DN pairs representing the lower and upper limits of the inclusive DN ranges, one pair per `DnRangeData` element) can be provided in the response.

**NOTE 2:** “isOffNet” and “site” attribute values are returned as empty string for each `DnRangeData` and `DnData` element.

### 5.65.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getUnassignedNumber"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.65.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
```

```
<com.broadsoft.protocols.nsooss.Command
  commandType="getUnassignedNumber"
  loginId="admin"
  result="0"
  transactionId="1">
  <commanddataArray>
    <com.broadsoft.protocols.nsooss.UnassignedNumberData>
      <dNRangeArray>
        <com.broadsoft.protocols.nsooss.DnRangeData
          fromDn="12404455550"
          isOffNet=""
          site=""
          toDn="+12404455555">
        </com.broadsoft.protocols.nsooss.DnRangeData>
        <com.broadsoft.protocols.nsooss.DnRangeData
          fromDn="12404457777"
          isOffNet=""
          site=""
          toDn="+12404457779">
        </com.broadsoft.protocols.nsooss.DnRangeData>
      </dNRangeArray>
    <dNArray>
      <com.broadsoft.protocols.nsooss.DnData
        dn="12404451111"
        isOffNet=""
        site="">
      </com.broadsoft.protocols.nsooss.DnData>
      <com.broadsoft.protocols.nsooss.DnData
        dn="12404450000"
        isOffNet=""
        site="">
      </com.broadsoft.protocols.nsooss.DnData>
    </dNArray>
  </com.broadsoft.protocols.nsooss.UnassignedNumberData>
</commanddataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.66 getResourceNes

### 5.66.1 Description

This command allows a system provider to retrieve all resource NEs currently provisioned in the system. All resource NEs, including Media Servers, SIMPLE, and UVS servers are returned.

**NOTE 1:** The “defaultCost” and “defaultWeight” attributes can be ignored if the attribute “default” is set to “false”.

**NOTE 2:** Attribute “type” can be set to “Media Server”, “SIMPLE server”, or “UVS”.

**NOTE 3:** Attribute “service” can be ignored if the attribute “type” is set to “SIMPLE server” because “service” applies only to Media Servers. Attribute “service” lists all the media services that a Media Server supports when used as a default Media Server by the Network Server.

### 5.66.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getResourceNes"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.66.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getResourceNes"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.ResourceNeData
        default="false"
        defaultCost="1"
        defaultWeight="1"
        location=""
        name="ms1"
        poll="false"
        service="ivr, conf"
        state="OnLine"
        staticCost="1"
        staticWeight="2"
        type="Media Server">
        <addressArray/>
      </com.broadsoft.protocols.nsooss.ResourceNeData>
      <com.broadsoft.protocols.nsooss.ResourceNeData
        default="false"
        defaultCost="1"
        defaultWeight="1"
        location="1514697"
        name="uvs1"
        poll="true"
        service="cfw-collab"
        state="OnLine"
        staticCost="9"
        staticWeight="99"
        type="UVS">
        <addressArray/>
      </com.broadsoft.protocols.nsooss.ResourceNeData>
      <com.broadsoft.protocols.nsooss.ResourceNeData
        default="true"
        defaultCost="4"
        defaultWeight="44"
        location="1514697"
        name="ms3"
```

```
        poll="true"
        service="ivr"
        state="OffLine"
        staticCost="7"
        staticWeight="77"
        type="Media Server">
        <addressArray/>
    </com.broadsoft.protocols.nsooss.ResourceNeData>
    <com.broadsoft.protocols.nsooss.ResourceNeData
        default="false"
        defaultCost="1"
        defaultWeight="1"
        location="1240364"
        name="simple1"
        poll="true"
        service=""
        state="OnLine"
        staticCost="3"
        staticWeight="33"
        type="SIMPLE Server">
        <addressArray/>
    </com.broadsoft.protocols.nsooss.ResourceNeData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.67 getResourceNeAdrrs

### 5.67.1 Description

This command allows a system provider to retrieve all addresses for a resource NE specified in the request.

**NOTE:** Only one resource NE can be retrieved per ResourceNeData command element. To retrieve the addresses of additional Resource NEs, use additional ResourceNeData command elements.

### 5.67.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getResourceNeAdrrs"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.ResourceNeData
        name="ms2">
      </com.broadsoft.protocols.nsooss.ResourceNeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.67.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getResourceNeAdrs"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.ResourceNeData
        default="false"
        defaultCost="1"
        defaultWeight="1"
        location="1514697"
        name="ms2"
        poll="true"
        service="ivr,conf, liveAudio"
        state="OffLine"
        staticCost="9"
        staticWeight="99"
        type="Media Server">
        <addressArray>
          <com.broadsoft.protocols.nsooss.ResourceNeAddrData
            address="2.2.2.2"
            cost="1"
            port="5679"
            transport="udp"
            weight="1"/>
          <com.broadsoft.protocols.nsooss.ResourceNeAddrData
            address="2.2.2.3"
            cost="1"
            port="5679"
            transport="tcp"
            weight="50"/>
          <com.broadsoft.protocols.nsooss.ResourceNeAddrData
            address="2.2.2.4"
            cost="2"
            port="5679"
            transport="unspecified"
            weight="50"/>
        </addressArray>
      </com.broadsoft.protocols.nsooss.ResourceNeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.68 addWebServerFarm

### 5.68.1 Description

This command allows a service provider to create a new Web Server Farm in the system. The Web Server Farm name cannot be used by another Web Server Farm in the system.

**NOTE:** Only one Web Server Farm can be added per addWebServerFarm command element. To add additional Web Server Farms, use additional addWebServerFarm command elements.

## 5.68.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addWebServerFarm"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmData
        name="WSF1"
        description="Web Server Farm number 1">
      </com.broadsoft.protocols.nsooss.WebServerFarmData >
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.68.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addWebServerFarm"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.69 deleteWebServerFarm

### 5.69.1 Description

This command allows a service provider to delete an existing Web Server Farm from the system. To use this command the Web Server Farm must be defined in the system.

**NOTE 1:** All attributes other than "name" specified in the request are ignored.

**NOTE 2:** Only one Web Server Farm can be deleted per deleteWebServerFarm command element. To delete additional Web Server Farms, use additional deleteWebServerFarm command elements.

### 5.69.2 Sample Input XML Structure

```
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteWebServerFarm"
```

```
loginId="admin"
transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsoss.WebServerFarmData
      name="WSF1">
    </com.broadsoft.protocols.nsoss.WebServerFarmData>
  </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.69.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteWebServerFarm"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.70 modifyWebServerFarm

### 5.70.1 Description

This command allows a service provider to modify the properties of an existing Web Server Farm defined in the system.

**NOTE:** Only one Web Server Farm can be modified per modifyWebServerFarm command element. To modify additional Web Server Farms, use additional modifyWebServerFarm command elements.

### 5.70.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyWebServerFarm"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.WebServerFarmData
        name="WSF1"
        description="Another description">
      </com.broadsoft.protocols.nsoss.WebServerFarmData >
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
```

```
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.70.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyWebServerFarm"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument
```

## 5.71 getWebServerFarms

### 5.71.1 Description

This command allows a service provider to retrieve the properties of all existing Web Server Farms defined in the system.

### 5.71.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarms"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.71.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarms"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmData
        description="Web Server Farm number 2"
        name="WSF2">
```



```
</com.broadsoft.protocols.nsooss.WebServerFarmData>
<com.broadsoft.protocols.nsooss.WebServerFarmData
  description="Another description"
  name="WSF1">
</com.broadsoft.protocols.nsooss.WebServerFarmData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.72 getWebServerFarm

### 5.72.1 Description

This command allows a service provider to retrieve the properties of an existing Web Server Farm defined in the system. The name of the Web Server Farm is required in the request and all other attributes are ignored when specified in the input XML string.

**NOTE:** Only one Web Server Farm can be retrieved per getWebServerFarm command element. To retrieve additional Web Server Farms, use additional getWebServerFarm command elements.

### 5.72.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarm"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmData
        name="WSF1">
      </com.broadsoft.protocols.nsooss.WebServerFarmData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.72.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarm"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmData
        description="Another description"
        name="WSF1">
    </com.broadsoft.protocols.nsooss.WebServerFarmData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

```
</com.broadsoft.protocols.nsoss.WebServerFarmData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.73 addWebServerFarmNode

### 5.73.1 Description

This command allows a service provider to define a new node for an existing Web Server Farm in the system.

**NOTE 1:** Node 0 is automatically created when provisioning a new Web Server Farm.

**NOTE 2:** The “name” must refer to a valid Web Server Farm provisioned in the system.

**NOTE 3:** The “nodeId” must not be used by an existing node on the Web Server Farm. The “nodeId” is a number between 0 and 9.

**NOTE 4:** If the “description” attribute is not specified in the request, the Network Server assumes that the new Web Server Farm node does not have a description.

**NOTE 5:** If the “isBrandingMaster” attribute is not specified in the request, the Network Server assumes that the new Web Server Farm node is not a branding server.

**NOTE 6:** If the field *version*, *patchLevel*, and *minimumPatchLevel* are not specified in the request, the Network Server initializes them to <unspecified>. These fields will be updated the next time the Web Server node tries to connect to the Network Server through the location API.

**NOTE 7:** Only one Web Server Farm node can be added per addWebServerFarmNode command element. To add additional Web Server Farm nodes, use additional addWebServerFarmNode command elements.

### 5.73.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="addWebServerFarmNode"
loginId="admin"
transactionId="1">
<commandDataArray>
<com.broadsoft.protocols.nsoss.WebServerFarmNodeData
name="WSF1"
nodeId="1"
description="Web Server Farm node 1"
isBrandingMaster="true"

state="OnLine">
</com.broadsoft.protocols.nsoss.WebServerFarmNodeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
```

```
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.73.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addWebServerFarmNode"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.74 deleteWebServerFarmNode

### 5.74.1 Description

This command allows a service provider to delete an existing Web Server Farm node from the system. To use this command, the Web Server Farm node must be defined in the system.

**NOTE 1:** Attributes other than "name" and "nodeId" are ignored if specified in the request.

**NOTE 2:** Only one Web Server Farm node can be deleted per deleteWebServerFarmNode command element. To delete additional Web Server Farm nodes, use additional deleteWebServerFarmNode command elements.

### 5.74.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteWebServerFarmNode"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.WebServerFarmNodeData
        name="WSF1"
        nodeId="1">
      </com.broadsoft.protocols.nsoss.WebServerFarmNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.74.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteWebServerFarmNode"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.75 modifyWebServerFarmNode

### 5.75.1 Description

This command allows a service provider to modify an existing Web Server Farm node defined in the system. The Web Server Farm node must be defined in the system.

**NOTE 1:** If the “description” attribute is not specified in the request, the Network Server assumes that the new Web Server Farm node does not have a description.

**NOTE 2:** If the “isBrandingMaster” attribute is not specified in the request, the Network Server assumes that the new Web Server Farm node is not a branding server.

**NOTE 3:** If the field *version*, *patchLevel*, and *minimumPatchLevel* are not specified in the request, the Network Server initializes them to <unspecified>. These fields are updated the next time the Web Server node tries to connect to the Network Server through the location API.

**NOTE 4:** Only one Web Server Farm node can be modified per `modifyWebServerFarmNode` command element. To modify additional Web Server Farm nodes, use additional `modifyWebServerFarmNode` command elements.

### 5.75.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyWebServerFarmNode"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.WebServerFarmNodeData
        name="WSF1"
        nodeId="1"
        version="Rel11.1b131"
        isBrandingMaster="true"
        patchLevel="MP3"
        minimumPatchLevel="MP3"
        description="Web Server Farm node 1"
```

```
        state="OffLine">
    </com.broadsoft.protocols.nsooss.WebServerFarmNodeData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.75.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyWebServerFarmNode"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.76 getWebServerFarmNodes

### 5.76.1 Description

This command allows a service provider to retrieve all existing Web Server Farm nodes defined in the system. The addresses defined for each node are also returned.

### 5.76.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarmNodes"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.76.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarmNodes"
    loginId="admin"
    result="0"
    transactionId="1">
```

```
<commandDataArray>
  <com.broadsoft.protocols.nsooss.WebServerFarmNodeData
    description="Node 0 (automatically added)"
    isBrandingMaster="true"
    name="WSF1"
    nodeId="0"
    patchLevel="unspecified"
    minimumPatchLevel="unspecified"
    state="OnLine"
    version="unspecified">
    <addressArray>
    </addressArray>
  </com.broadsoft.protocols.nsooss.WebServerFarmNodeData>
  <com.broadsoft.protocols.nsooss.WebServerFarmNodeData
    description="Web Server Farm node 1"
    isBrandingMaster="true"
    name="WSF1"
    nodeId="1"
    patchLevel="MP3"
    minimumPatchLevel="MP3"
    state="OffLine"
    version="Rel11.1b131">
    <addressArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmAddrData
        address="1.1.1.1"
        cost="1"
        type="Access"
        weight="50">
      </com.broadsoft.protocols.nsooss.WebServerFarmAddrData>
    </addressArray>
  </com.broadsoft.protocols.nsooss.WebServerFarmNodeData>
  <com.broadsoft.protocols.nsooss.WebServerFarmNodeData
    description="Node 0 (automatically added)"
    isBrandingMaster="true"
    name="WSF2"
    nodeId="0"
    patchLevel="unspecified"
    minimumPatchLevel="unspecified"
    state="OnLine"
    version="unspecified">
    <addressArray>
    </addressArray>
  </com.broadsoft.protocols.nsooss.WebServerFarmNodeData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.77 getWebServerFarmNode

### 5.77.1 Description

This command allows a service provider to retrieve the properties of an existing web server farm node defined in the system. The “name” and “nodeId” of the Web Server Farm node are required in the request and all other attributes are ignored when specified in the input XML string.

**NOTE:** Only one Web Server Farm node can be retrieved per getWebServerFarmNode command element. To retrieve additional Web Server Farm nodes, use additional getWebServerFarmNode command elements.

### 5.77.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarmNode"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmNodeData
        name="WSF1"
        nodeId="1">
      </com.broadsoft.protocols.nsooss.WebServerFarmNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.77.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarmNodes"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmNodeData
        description="Web Server Farm node 1"
        isBrandingMaster="true"
        name="WSF1"
        nodeId="1"
        patchLevel="MP3"
        minimumPatchLevel="MP3"
        state="OffLine"
        version="Rel11.1b131">
        <addressArray>
          <com.broadsoft.protocols.nsooss.WebServerFarmAddrData
            address="1.1.1.1"
            cost="1"
          </com.broadsoft.protocols.nsooss.WebServerFarmAddrData>
        </addressArray>
      </com.broadsoft.protocols.nsooss.WebServerFarmNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

```
        type="Access"
        weight="50">
      </com.broadsoft.protocols.nsoss.WebServerFarmAddrData>
    </addressArray>
  </com.broadsoft.protocols.nsoss.WebServerFarmNodeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.78 addWebServerFarmAddr

### 5.78.1 Description

This command allows a service provider to define a new address for an existing Web Server Farm node in the system.

**NOTE 1:** The “name” and “nodeId” must refer to a valid Web Server Farm node provisioned in the system.

**NOTE 2:** Although the DTD cannot capture this rule, the “type” of address is required and must be provided. Valid types are defined in the DTD. If the type is “Alias”, then the attributes “cost” and “weight” are ignored by the Network Server if they are specified in the XML request. The Network Server uses predefined values for cost and weight when the address type is set to “Alias”. If the type is “Access”, then the attributes “cost” and “weight” must be provided.

**NOTE 3:** Only one Web Server Farm address can be added per addWebServerFarmAddr command element. To add additional Web Server Farm addresses, use additional addWebServerFarmAddr command elements.

### 5.78.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsoss.Command
      commandType="addWebServerFarmAddr"
      loginId="admin"
      transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsoss.WebServerFarmNodeData
          name="WSF1"
          nodeId="1">
          <addressArray>
            <com.broadsoft.protocols.nsoss.WebServerFarmAddrData
              address="192.168.12.5"
              type="Access"
              cost="1"
              weight="99">
            </com.broadsoft.protocols.nsoss.WebServerFarmAddrData>
          </addressArray>
        </com.broadsoft.protocols.nsoss.WebServerFarmNodeData>
      </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
  </commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```



### 5.78.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addWebServerFarmAddr"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.79 deleteWebServerFarmAddr

### 5.79.1 Description

This command allows a service provider to delete an existing Web Server Farm node address from the system. To use this command, the Web Server Farm node address must be defined in the system.

**NOTE 1:** Attributes other than “name”, “nodeId”, and “address” are ignored if specified in the request.

**NOTE 2:** Only one Web Server Farm node address can be deleted per deleteWebServerFarmAddr command element. To delete additional Web Server Farm node addresses, use additional deleteWebServerFarmAddr command elements.

### 5.79.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteWebServerFarmAddr"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.WebServerFarmNodeData
        name="WSF1"
        nodeId="1">
        <addressArray>
          <com.broadsoft.protocols.nsoss.WebServerFarmAddrData
            address="192.168.12.5">
          </com.broadsoft.protocols.nsoss.WebServerFarmAddrData>
        </addressArray>
      </com.broadsoft.protocols.nsoss.WebServerFarmNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.79.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteWebServerFarmAddr"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.80 modifyWebServerFarmAddr

### 5.80.1 Description

This command allows a service provider to modify an existing Web Server Farm node address defined in the system. To use this command, the hosting NE node address must be defined in the system.

**NOTE 1:** The “name” and “nodeId” must refer to a valid Web Server Farm node provisioned in the system.

**NOTE 2:** Although the DTD cannot capture this rule, the “type” of address is required and must be provided. Valid types are defined in the DTD. If the type is “Alias”, then the attributes “cost” and “weight” are ignored by the Network Server if they are specified in the XML request. The Network Server uses predefined values for cost and weight when the address type is set to “Alias”. If the type is “Access”, then the attributes “cost” and “weight” must be provided.

**NOTE 3:** Only one Web Server Farm address can be modified per modifyWebServerFarmAddr command element. To modify additional Web Server Farm addresses, use additional modifyWebServerFarmAddr command elements.

### 5.80.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyWebServerFarmAddr"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.WebServerFarmNodeData
        name="WSF1"
        nodeId="1">
        <addressArray>
          <com.broadsoft.protocols.nsoss.WebServerFarmAddrData
            address="192.168.12.5"
            cost="2"
```

```
        type="Access"
        weight="60">
    </com.broadsoft.protocols.nsoss.WebServerFarmAddrData>
</addressArray>
</com.broadsoft.protocols.nsoss.WebServerFarmNodeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.80.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyWebServerFarmAddr"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.81 getWebServerFarmAddr

### 5.81.1 Description

This command allows a service provider to retrieve all addresses defined for an existing Web Server Farm node.

### 5.81.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getWebServerFarmAddr"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.WebServerFarmNodeData
        name="WSF1"
        nodeId="1">
      </com.broadsoft.protocols.nsoss.WebServerFarmNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.81.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getWebServerFarmAddrs"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.WebServerFarmNodeData
        description="Web Server Farm node 1"
        isBrandingMaster="true"
        name="WSF1"
        nodeId="1"
        patchLevel="MP3"
        minimumPatchLevel="MP3"
        state="OffLine"
        version="Rel11.1b131">
        <addressArray>
          <com.broadsoft.protocols.nsooss.WebServerFarmAddrData
            address="1.1.1.1"
            cost="1"
            type="Access"
            weight="50">
          </com.broadsoft.protocols.nsooss.WebServerFarmAddrData>
          <com.broadsoft.protocols.nsooss.WebServerFarmAddrData
            address="192.168.12.5"
            cost="2"
            type="Alias"
            weight="60">
          </com.broadsoft.protocols.nsooss.WebServerFarmAddrData>
        </addressArray>
      </com.broadsoft.protocols.nsooss.WebServerFarmNodeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.82 getAllSystemMediaCodec

### 5.82.1 Description

This command allows a system provider to query the Network Server for all the mediaCodecs currently defined in the system that can be assigned to network elements. A mediaCodec is a system tag that defines the media capability of a network element. A mediaCodec might be used to identify an entire media type (such as audio or video) or a specific codec (encoding name) within a media type (such as H263-2000 in video or PCMU in audio).

**NOTE:** The command takes no parameter. It returns the list of all mediaCodecs currently defined in the system.

### 5.82.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="getAllSystemMediaCodec"
      loginId="admin"
      transactionId="1">
    </com.broadsoft.protocols.nsooss.Command>
  </commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.82.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="getAllSystemMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="alaw">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="audio">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="g726-32">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="g729a">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="h263-1998">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="h263-2000">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="h264">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="ulaw">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
      <com.broadsoft.protocols.nsooss.SystemMediaCodecData
        mediaCodec="video">
      </com.broadsoft.protocols.nsooss.SystemMediaCodecData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.83 addNetworkElementMediaCodec

### 5.83.1 Description

This command allows a system provider to add one or more mediaCodecs to the existing list of mediaCodecs that a network element supports.

**NOTE 1:** The attribute “neName” is required and it must identify an existing network element of type “neType” in the system.

**NOTE 2:** The attribute “neType” is required and it must specify the type of network element being modified. Valid network element types are “hosting” (hosting NE), “routing” (routing NE), or “resource” (Resource NE).

**NOTE 3:** The attribute “codecBasedRtgSupported” is optional in a request. If set to “true”, the network element supports codec-based routing. As a result, the Network Server may re-order contacts in a call processing response by comparing the call’s SDP (in the incoming SIP INVITE message) with the mediaCodecs that the network element actually supports. If set to “false”, codec-based routing is disabled for this network element so contacts pointing to this network element will not be shuffled based on mediaCodec level of support.

**NOTE 4:** The “mediaCodecArray” contains the mediaCodecs to be added to the list of mediaCodecs that the network element should support. For this operation to succeed, the provided mediaCodecs should already exist in the system. The operation will fail if one or more mediaCodecs is already in the list of supported mediaCodecs for the specified network element.

**NOTE 5:** Only one network element can be modified per addNetworkElementMediaCodec command element. To modify additional network elements, use additional addNetworkElementMediaCodec command elements.

### 5.83.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="addNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsooss.NetworkElementMediaCodecData
          codecBasedRtgSupported="true"
          neName="NGW"
          neType="routing">
          <mediaCodecArray>
            <com.broadsoft.protocols.nsooss.MediaCodecData
              mediaCodecName="audio">
            </MediaCodecData>
            <com.broadsoft.protocols.nsooss.MediaCodecData
              mediaCodecName="h263-2000"
```

```
        priority="20">
      </MediaCodecData>
    </mediaCodecArray>
  </com.broadsoft.protocols.nsooss.NetworkElementMediaCodecData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.83.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="addNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
      </commandDataArray>
    </com.broadsoft.protocols.nsooss.Command>
  </commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.84 deleteNetworkElementMediaCodec

### 5.84.1 Description

This command allows a system provider to delete one or more mediaCodecs from the existing list of mediaCodecs that a network element supports.

**NOTE 1:** The attribute "neName" is required and it must identify an existing network element of type "neType" in the system.

**NOTE 2:** The attribute "neType" is required and it must specify the type of network element being modified. Valid network element types are "hosting" (hosting NE), "routing" (routing NE), or "resource" (Resource NE).

**NOTE 3:** The attribute "codecBasedRtgSupported" is optional in a request. If set to "true", the network element supports codec-based routing. As a result, the Network Server may re-order contacts in a call processing response by comparing the call's SDP (in the incoming SIP INVITE message) with the mediaCodecs that the network element actually supports. If set to "false", codec-based routing is disabled for this network element so contacts pointing to this network element will not be shuffled based on mediaCodec level of support.

**NOTE 4:** The "mediaCodecArray" contains the mediaCodecs to be removed from the list of mediaCodecs that the network element should support. For this operation to succeed, the provided mediaCodecs should already exist in the system. The operation will fail if one or more mediaCodecs is already not in the list of supported mediaCodecs for the specified network element.

**NOTE 5:** Only one network element can be modified per deleteNetworkElementMediaCodec command element. To modify additional network elements, use additional deleteNetworkElementMediaCodec command elements.

### 5.84.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="deleteNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsooss.NetworkElementMediaCodecData
          codecBasedRtgSupported="false"
          neName="NGW"
          neType="routing">
          <mediaCodecArray>
            <com.broadsoft.protocols.nsooss.MediaCodecData
              mediaCodecName="ulaw"
              priority="45">
            </MediaCodecData>
          </mediaCodecArray>
        </com.broadsoft.protocols.nsooss.NetworkElementMediaCodecData>
      </commandDataArray>
    </com.broadsoft.protocols.nsooss.Command>
  </commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.84.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="deleteNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
    </commandDataArray>
    </com.broadsoft.protocols.nsooss.Command>
  </commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```



## 5.85 modifyNetworkElementMediaCodec

### 5.85.1 Description

This command allows a system provider to reset the list of mediaCodecs that a network element supports. In doing so, all mediaCodecs that are present before the command is run are deleted from the network element list and replaced by the mediaCodecs that are listed in the command.

**NOTE 1:** The attribute “neName” is required and it must identify an existing network element of type “neType” in the system.

**NOTE 2:** The attribute “neType” is required and it must specify the type of network element being modified. Valid network element types are “hosting” (hosting NE), “routing” (routing NE), or “resource” (Resource NE).

**NOTE 3:** The attribute “codecBasedRtgSupported” is optional in a request. If set to “true”, the network element supports codec-based routing. As a result, the Network Server may re-order contacts in a call processing response by comparing the call’s SDP (in the incoming SIP INVITE message) with the mediaCodecs that the network element actually supports. If set to “false”, codec-based routing is disabled for this network element so contacts pointing to this network element will not be shuffled based on mediaCodec level of support.

**NOTE 4:** The “mediaCodecArray” contains the exact list of mediaCodecs that the network element should now support. All mediaCodecs previously assigned to the network element are first deleted.

**NOTE 5:** Only one network element can be modified per modifyNetworkElementMediaCodec command element. To modify additional network elements, use additional modifyNetworkElementMediaCodec command elements.

### 5.85.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="modifyNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsooss.NetworkElementMediaCodecData
          codecBasedRtgSupported="true"
          neName="NGW"
          neType="routing">
          <mediaCodecArray>
            <com.broadsoft.protocols.nsooss.MediaCodecData
              mediaCodecName="audio">
            </MediaCodecData>
            <com.broadsoft.protocols.nsooss.MediaCodecData
              mediaCodecName="video">
            </MediaCodecData>
```

```
        </mediaCodecArray>
    </com.broadsoft.protocols.nsooss.NetworkElementMediaCodecData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.85.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="modifyNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
      </commandDataArray>
    </com.broadsoft.protocols.nsooss.Command>
  </commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.86 getNetworkElementMediaCodec

### 5.86.1 Description

This command allows a system provider to retrieve, for a network element, the list of all mediaCodecs that this network element supports as well as an indication whether codec-based routing is enabled or not.

**NOTE 1:** The attribute “neName” is required and it must identify an existing network element of type “neType” in the system.

**NOTE 2:** The attribute “neType” is required and it must specify the type of network element being queried. Valid network element types are “hosting” (hosting NE), “routing” (routing NE), or “resource” (Resource NE).

**NOTE 3:** The attribute “codecBasedRtgSupported” is part of the response (ignored if included in the request). If set to “true”, the network element supports codec-based routing. As a result, the Network Server may re-order contacts in a call processing response by comparing the call’s SDP (in the incoming SIP INVITE message) with the mediaCodecs that the network element actually supports. If set to “false”, codec-based routing is disabled for this network element so contacts pointing to this network element will not be shuffled based on mediaCodec level of support.

**NOTE 4:** The “mediaCodecArray” included in the response contains the exact list of mediaCodecs that the network element supports.

**NOTE 5:** Only one network element can be queried per `getNetworkElementMediaCodec` command element. To query additional network elements, use additional `getNetworkElementMediaCodec` command elements.

### 5.86.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsoss.Command
      commandType="getNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsoss.NetworkElementMediaCodecData
          neName="NGW"
          neType="routing">
        </com.broadsoft.protocols.nsoss.NetworkElementMediaCodecData>
      </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
  </commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.86.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument
  protocol="NSOSS"
  version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsoss.Command
      commandType="getNetworkElementMediaCodec"
      loginId="admin"
      elementStartIndex="-1"
      numberOfElements="-1"
      result="0"
      transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsoss.NetworkElementMediaCodecData
          codecBasedRtgSupported="true"
          neName="NGW"
          neType="routing">
          <mediaCodecArray>
            <com.broadsoft.protocols.nsoss.MediaCodecData
              mediaCodecName="audio">
            </MediaCodecData>
            <com.broadsoft.protocols.nsoss.MediaCodecData
              mediaCodecName="video">
            </MediaCodecData>
          </mediaCodecArray>
        </com.broadsoft.protocols.nsoss.NetworkElementMediaCodecData>
      </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
  </commandArray>
```

```
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.87 addDefaultSourceId

### 5.87.1 Description

This command is used to add a new default sourceId mapping range to the Network Server.

### 5.87.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addDefaultSourceId"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.DefaultSourceIdData
        fromDn="1514333"
        toDn="1514333"
        sourceid="12345">
      </com.broadsoft.protocols.nsooss.DefaultSourceIdData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.87.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addDefaultSourceId"
    loginId="admin"
    result="0"
    transactionid="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.88 modifyDefaultSourceId

### 5.88.1 Description

This command is used to modify a default sourceId mapping range that is already defined on the Network Server.

### 5.88.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyDefaultSourceId"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.DefaultSourceIdData
        fromDn="1514333"
        toDn="1514333"
        sourceid="12345">
      </com.broadsoft.protocols.nsooss.DefaultSourceIdData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.88.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyDefaultSourceId"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.89 deleteDefaultSourceId

### 5.89.1 Description

This command is used to delete an existing default sourceId mapping range from the Network Server.

### 5.89.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
```

```
<com.broadsoft.protocols.nsooss.Command
  commandType="deleteDefaultSourceId"
  loginId="admin"
  transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsooss.DefaultSourceIdData
      fromDn="1514333"
      toDn="1514333">
    </com.broadsoft.protocols.nsooss.DefaultSourceIdData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.89.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteDefaultSourceId"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.90 getDefaulSourceIds

### 5.90.1 Description

This command is used to view all default sourceId mapping ranges from the Network Server.

### 5.90.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getDefaulSourceIds"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.90.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
```

```
<com.broadsoft.protocols.nsooss.Command
  commandType="getDefaultSourceIds"
  loginId="admin"
  result="0"
  transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsooss.DefaultSourceIdData
      fromDn="1514333"
      toDn="1514333"
      sourceid="12345">
    </com.broadsoft.protocols.nsooss.DefaultSourceIdData>
    <com.broadsoft.protocols.nsooss.DefaultSourceIdData
      fromDn="1514697"
      toDn="1514697"
      sourceid="562367">
    </com.broadsoft.protocols.nsooss.DefaultSourceIdData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.91 addLinePort

### 5.91.1 Description

This command allows a service provider to add a new line/port in the system and to associate it with an existing user identified by the user's login ID alias.

**NOTE 1:** The *loginIdUrl* attribute is required and should refer to an existing login ID alias in the system.

**NOTE 2:** The *linePort* attribute is required and should identify a new line/port to be added in the system. If the line/port already exists in the system, the command fails. The line/port is assigned to the login ID alias specified in the request.

**NOTE 3:** The *supportsOrigDiversion* attribute is optional. If specified and set to "true", it flags this line/port as being a PBX integration line/port, meaning that the Application Server can accept originations from this line/port when it is the diverting party in a PBX deflection call scenario.

### 5.91.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addLinePort"
    ... loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.LinePortData
        linePort="15145559999@sp1.com"
        loginIdUrl="johndoe@broadsoft.com"
        supportsOrigDiversion="false">
      </com.broadsoft.protocols.nsooss.LinePortData>
    </commandDataArray>
```

```
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.91.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsoss.Command
      commandType="addLinePort"
      loginId="admin"
      result="0"
      transactionId="1">
      <commandDataArray>
      </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
  </commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.92 deleteLinePort

### 5.92.1 Description

This command allows a service provider to delete an existing line/port from the system and to remove all associations that this line/port has with users in the system.

**NOTE:** The *linePort* attribute is required and should identify an existing line/port in the system. If the line/port does not exist in the system, the command fails.

### 5.92.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
  <commandArray>
    <com.broadsoft.protocols.nsoss.Command
      commandType="deleteLinePort"
      loginId="admin"
      transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsoss.LinePortData
          linePort="1514555999@sp1.com">
        </com.broadsoft.protocols.nsoss.LinePortData>
      </commandDataArray>
    </com.broadsoft.protocols.nsoss.Command>
  </commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.92.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
  <commandArray>
```



```
<com.broadsoft.protocols.nsooss.Command
  commandType="deleteLinePort"
  loginId="admin"
  result="0"
  transactionId="1">
  <commandDataArray>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.93 modifyLinePort

### 5.93.1 Description

This command allows a service provider to replace a specific line/port assigned to a user by a new line/port assigned to the same user. The line/port to be replaced (the old one) needs to be specified. The command can also be used to change the value of line/port attributes. Currently only *supportsOrigDiversion* can be changed.

**NOTE 1:** The *newLinePort* attribute is optional. If present, it should identify a new line/port to be added in the system. If the line/port already exists in the system, the command fails. The new line/port is assigned to the login ID alias currently owning *linePort*.

**NOTE 2:** The *linePort* attribute is required and should identify an existing line/port in the system. If the line/port does not exist in the system, the command fails.

**NOTE 3:** The *supportsOrigDiversion* attribute is optional. If specified and set to "true", it flags the line/port (*newLinePort* if present; otherwise, *linePort*) as being a PBX integration line/port, meaning that the Application Server can accept originations from this line/port when it is the diverting party in a PBX deflection call scenario. If specified and set to "false", it flags the line/port (*newLinePort* if present; otherwise, *linePort*) as being a non-PBX integration line/port.

### 5.93.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyLinePort"
    ... loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.LinePortData
        linePort="15145559999@sp1.com"
        newLinePort="1514558888@sp2.com"
        supportsOrigDiversion="true">
      </com.broadsoft.protocols.nsooss.LinePortData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.93.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyDefaultSourceId"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.94 getLinePorts

### 5.94.1 Description

This command allows a service provider to get one or more line/ports in the system. This command returns line/ports owned by users.

The OSS interface limits the number of line/ports that are returned within a command response. If the OSS client wants a complete response, it needs to continue sending get requests to the Network Server until all line/ports are received. To do this, the client provides two attributes, *elementStartIndex*, and *numberOfElements*. Attribute *elementStartIndex* specifies the index of the first element that it wants to be returned, with index "0" being the first element. Attribute *numberOfElements* specifies how many elements it wants to be returned, including the element at *elementStartIndex*. If the OSS client does not use these attributes, the Network Server never returns more than 100 line/ports.

**NOTE 1:** To get all line/ports assigned to users hosted on a specific hosting NE, the *hostingNE* attribute should contain the name of an existing hosting NE in the system.

**NOTE 2:** To get all line/ports assigned to users all part of a specific group, the *hostingNE* attribute should contain the name of an existing hosting NE in the system and the *groupID* attribute should contain the name of an existing group in the system.

**NOTE 3:** To get all line/ports assigned to a specific user, the *loginIdUrl* attribute should contain the login ID alias of the user.

**NOTE 4:** To get all line/ports defined in the system, no attribute needs to be set.

**NOTE 5:** The response contains one line/port per LinePortData element.

### 5.94.2 Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getLinePorts"
  >
```

```
loginId="admin"  
elementStartIndex="0"  
numberOfElements="50"  
transactionId="1">  
<commanddataArray>  
  <com.broadsoft.protocols.nsoss.LinePortData  
    loginIdUrl="johndoe@broadsoft.com">  
  </com.broadsoft.protocols.nsoss.LinePortData>  
</commanddataArray>  
</com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.94.3 Sample Output XML

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
  version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="getLinePorts"  
    loginId="admin"  
    elementStartIndex="0"  
    numberOfElements="50"  
    result="0"  
    transactionId="1">  
    <commanddataArray>  
      <com.broadsoft.protocols.nsoss.LinePortData  
        groupId="group1"  
        hostingNE="host1"  
        linePort="15145559999@sp1.com"  
        loginIdUrl="johndoe@broadsoft.com"  
        supportsOrigDiversion="false">  
      </com.broadsoft.protocols.nsoss.LinePortData>  
      <com.broadsoft.protocols.nsoss.LinePortData  
        groupId="group1"  
        hostingNE="host1"  
        linePort="12405556789@sp2.com"  
        loginIdUrl="johndoe@broadsoft.com"  
        supportsOrigDiversion="false">  
      </com.broadsoft.protocols.nsoss.LinePortData>  
    </commanddataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</commandArray>
```

## 5.95 addNetworkTranslationIndex

### 5.95.1 Description

This command allows a service provider to create a network translation index. If the network translation index exists, the operation fails.

**NOTE 1:** A network translation index can be created with no attributes.

**NOTE 2:** If specified, the enterprise must refer to an existing enterprise in the system. The enterprise can be defaulted to NIL\_ENTERPRISE.

**NOTE 3:** If specified, the site must refer to a valid site provisioned in the enterprise identified by the enterprise attribute. If it is not specified and an enterprise is specified, the site is defaulted to DFLT\_SITE.

**NOTE 4:** If specified, the lcaid must refer to an existing lcaid in the system. The lcaid can also be set to an empty string.

**NOTE 5:** If specified, the intraLataPic, interLataPic, or internationalPIC must refer to PICs in the system.

**NOTE 6:** If specified, the profile must refer to an existing profile in the system. The profile can also be set to "NIL\_PROFILE".

### 5.95.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addNetworkTranslationIndex"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
        nti="anNTI"
      </com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
```

### 5.95.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType=" addNetworkTranslationIndex "
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.96 modifyNetworkTranslationIndex

### 5.96.1 Description

This command allows a service provider to modify an existing network translation index. Although the DTD cannot capture this rule, the carrier, profile, defaultTranslationDN, lcaid, interLataPic, interLataPic and internationalPic information MUST be provided when modifying the enterprise's properties.

**NOTE 1:** A network translation index can be created with no attributes.

**NOTE 2:** If specified, the enterprise must refer to an existing enterprise in the system. The enterprise can be defaulted to NIL\_ENTERPRISE.

**NOTE 3:** If specified, the site must refer to a valid site provisioned in the enterprise identified by the enterprise attribute. If it is not specified and an enterprise is specified, the site is defaulted to DFLT\_SITE.

**NOTE 4:** If specified, the lcaid must refer to an existing lcaid in the system.

**NOTE 5:** If specified, the intraLataPic, interLataPic, or internationalPIC must refer to PICs in the system.

**NOTE 6:** Setting a field to an empty string automatically clears it.

### 5.96.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyNetworkTranslationIndex"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
        nti="anNTI"
        enterprise="NIL_ENTERPRISE"
        site="DFTL_SITE"
        interLataCarrier="NIL_CAC"
        internationalCarrier="NIL_CAC"
        intraLataCarrier="NIL_CAC"
        profile="Profile2"
        defaultTranslationDN=""
        lcaid="">
      </com.broadsoft.protocols.nsoss.NetworkTranslationIndexData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.96.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="modifyNetworkTranslationIndex"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.97 deleteNetworkTranslationIndex

### 5.97.1 Description

This command allows a service provider to delete an existing network translation index.

**NOTE:** Only one network translation index can be deleted per *deleteNetworkTranslationIndex* command element. To delete additional enterprises, use additional *deleteNetworkTranslationIndex* command elements.

### 5.97.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteNetworkTranslationIndex"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
        nti="anNTI">
      </com.broadsoft.protocols.nsoss.NetworkTranslationIndexData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.97.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteNetworkTranslationIndex"
    loginId="admin"
    result="0"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
```

```
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.98 getNetworkTranslationIndex

### 5.98.1 Description

This command allows a service provider to query the Network Server for an existing network translation index.

**NOTE:** Only one network translation index can be queried per `getNetworkTranslationIndex` command element. To query for additional enterprises, use additional `getNetworkTranslationIndexes` command elements.

### 5.98.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getNetworkTranslationIndex"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.NetworkTranslationIndexData
        nti="anNTI">
      </com.broadsoft.protocols.nsooss.NetworkTranslationIndexData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.98.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getNetworkTranslationIndex"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.NetworkTranslationIndexData
        nti="anNTI"
        enterprise="NIL_ENTERPRISE"
        site="DFTL_SITE"
        interLataCarrier="NIL_CAC"
        internationalCarrier="NIL_CAC"
        intraLataCarrier="NIL_CAC"
        profile="Profile2"
        defaultTranslationDN=""
        lcaid="">
      </com.broadsoft.protocols.nsooss.NetworkTranslationIndexData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
```

```
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.99 getNetworkTranslationIndexes

### 5.99.1 Description

This command allows a service provider to query the Network Server system for all existing network translation indexes.

### 5.99.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getNetworkTranslationIndexes"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.99.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getEnterprises"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
        nti="anNTI"
        enterprise="NIL_ENTERPRISE"
        site="DFTL_SITE"
        interLataCarrier="NIL_CAC"
        internationalCarrier="NIL_CAC"
        intraLataCarrier="NIL_CAC"
        profile="Profile2"
        defaultTranslationDN =""
        lcaid="">
      </com.broadsoft.protocols.nsoss.NetworkTranslationIndexData>
      <com.broadsoft.protocols.nsoss.NetworkTranslationIndexData
        nti="anotherNTI"
        enterprise="enterprise1"
        site="DFTL_SITE"
        interLataCarrier="NIL_CAC"
        internationalCarrier="NIL_CAC"
        intraLataCarrier="NIL_CAC"
        profile="systemProf"
        defaultTranslationDN ="1514333"
        lcaid="">
      </com.broadsoft.protocols.nsoss.NetworkTranslationIndexData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
```



```
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.100 getPeers

### 5.100.1 Description

This command is used to view peer Network Server-related attributes, such as the peer type and host name.

**NOTE:** When configuring replication, it is important to use the canonical host name of the server. The canonical host name can be retrieved by typing the *hostname* command without arguments. Note, however, that using the IP address or an alias causes replication to fail due to a TimesTen limitation.

### 5.100.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getPeers"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.100.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="getPeers"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
<com.broadsoft.protocols.nsooss.PeerData address="localhost"
elementStartIndex="-1"
hostname="localhost"
numberOfElements="-1"
state="unlocked"
```

## 5.101 addDataCenter

### 5.101.1 Description

This command is used to add a new data center instance to the system.

**NOTE:** The name is mandatory and must be unique.

### 5.101.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addDataCenter"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.DataCenterData
        name="aUniqueDataCenterName"
        description="a data center description">
      </com.broadsoft.protocols.nsoss.DataCenterData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.101.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command commandType="addDataCenter"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.102 deleteDataCenter

### 5.102.1 Description

This command is used to delete existing data center instances in the system.

**NOTE:** The name is mandatory and must be an existing Data Center.

### 5.102.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteDataCenter"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.DataCenterData
        name="aUniqueDataCenterName">
      </com.broadsoft.protocols.nsoss.DataCenterData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.102.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command commandType="deleteDataCenter"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.103 modifyDataCenter

### 5.103.1 Description

This command is used to modify existing data center instances in the system.

**NOTE 1:** The “name” must refer to a valid Data Center provisioned in the system.

**NOTE 2:** Only one Data Center can be modified per *modifyDataCenter* command element. To modify additional Data Centers, use additional *modifyDataCentercommand* elements.

### 5.103.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyDataCenter"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.DataCenterData
        name="aUniqueDataCenterName"
        description="a new description">
      </com.broadsoft.protocols.nsooss.DataCenterData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.103.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="modifyDataCenter"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.104 getDataCenter

### 5.104.1 Description

This command is used to view an existing data center instance in the system.

**NOTE:** The name is mandatory and must refer to an existing Data Center in the system.

### 5.104.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getDataCenter"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.DataCenterData
        name="aUniqueDataCenterName"
        description="">
      </com.broadsoft.protocols.nsoss.DataCenterData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.104.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command commandType="getDataCenter"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
<com.broadsoft.protocols.nsoss.DataCenterData description="a new
description"
elementStartIndex="-1"
name="aUniqueDataCenterName"
numberOfElements="-1"
>
</com.broadsoft.protocols.nsoss.DataCenterData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.105 getDataCenters

### 5.105.1 Description

This command is used to view all existing data centers instances in the system.

### 5.105.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getDataCenters"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.105.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="getDataCenters"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commanddataArray>
<com.broadsoft.protocols.nsooss.DataCenterData description="another
description"
elementStartIndex="-1"
name="anotherDataCenterName"
numberOfElements="-1"
>
</com.broadsoft.protocols.nsooss.DataCenterData>
<com.broadsoft.protocols.nsooss.DataCenterData description="a new
description"
elementStartIndex="-1"
name="aUniqueDataCenterName"
numberOfElements="-1"
>
</com.broadsoft.protocols.nsooss.DataCenterData>
</commanddataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.106 addHostingNESet

### 5.106.1 Description

This command allows a user to add a hosting NE set to the system. If the set to be added should also become the system set, (that is, the attribute *systemSet* has been set to "true"), then the set currently flagged as the system set (if any) is no longer the selected system set.

**NOTE:** The name is mandatory and must be unique.

### 5.106.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addHostingNESet"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNESetData
        name="aHostingSet3"
        systemSet="">
        <hostingNEArray>
          <string value="ASAus"/>
          <string value="ASsanity"/>
        </hostingNEArray>
      </com.broadsoft.protocols.nsooss.HostingNESetData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.106.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="addHostingNESet"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.107 deleteHostingNESet

### 5.107.1 Description

This command allows a user to delete a hosting NE set from the system. If the set to be deleted is flagged as being the system set, the CLI waits for a confirmation from the user before deleting the set.

**NOTE:** The name is mandatory and must refer to an existing *HostingNESet* in the system.

### 5.107.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteHostingNESet"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNESetData
        name="aHostingSet3"
        systemSet="">
      </com.broadsoft.protocols.nsoss.HostingNESetData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.107.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="deleteHostingNESet"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
```



## 5.108 modifyHostingNESet

### 5.108.1 Description

This command allows a user to change a hosting NE set in the system. If the set to be modified should become the system set, indicated with attribute *systemSet* set to "true", then the set currently flagged as being the system set (if any) is no longer the selected system set.

**NOTE:** The name is mandatory and must refer to an existing *HostingNESet* in the system.

### 5.108.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyHostingNESet"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNESetData
        name="aHostingSet2"
        systemSet="">
        <hostingNEArray>
          <string value="ASAus_mod"/>
          <string value="ASSanity_mod"/>
        </hostingNEArray>
      </com.broadsoft.protocols.nsooss.HostingNESetData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.108.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command
commandType="modifyHostingNESet"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.109 getHostingNESet

### 5.109.1 Description

This command allows a user to view a specific hosting NE set in the system.

**NOTE:** The name is mandatory and must refer to an existing *HostingNESet* in the system.

### 5.109.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="getHostingNESet"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.HostingNESetData
        name="aHostingSet2"
        systemSet="">
      </com.broadsoft.protocols.nsooss.HostingNESetData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.109.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="getHostingNESet"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
<com.broadsoft.protocols.nsooss.HostingNESetData elementStartIndex="-1"
name="aHostingSet2"
numberOfElements="-1"
systemSet="false"
>
<hostingNEArray>
<string value="testAS"
>
</string>
<string value="ASaus"
>
</string>
</hostingNEArray>
</com.broadsoft.protocols.nsooss.HostingNESetData>
</commandDataArray>
```

```
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument></commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.110 getHostingNESets

### 5.110.1 Description

This command allows a user to view all the hosting NE sets in the system.

### 5.110.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getHostingNESets"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.110.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command commandType="getHostingNESets"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
<com.broadsoft.protocols.nsoss.HostingNESetData elementStartIndex="-1"
name="aHostingSet"
numberOfElements="-1"
systemSet="false"
>
<hostingNEArray>
</hostingNEArray>
</com.broadsoft.protocols.nsoss.HostingNESetData>
<com.broadsoft.protocols.nsoss.HostingNESetData elementStartIndex="-1"
name="aHostingSet2"
numberOfElements="-1"
systemSet="false"
>
<hostingNEArray>
</hostingNEArray>
</com.broadsoft.protocols.nsoss.HostingNESetData>
</commandDataArray>
```

```
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.111 addHostingNEHostingNESet

### 5.111.1 Description

This command allows a user to add a hosting NE to a hosting NE set.

**NOTE:** The name is mandatory and must refer to an existing *HostingNESet* in the system.

### 5.111.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addHostingNEHostingNESet"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNESetData
        name="aHostingSet2"
        systemSet="">
        <hostingNEArray>
          <string value="testAS"/>
        </hostingNEArray>
      </com.broadsoft.protocols.nsoss.HostingNESetData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.111.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="addHostingNEHostingNESet"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.112 deleteHostingNEHostingNESet

### 5.112.1 Description

This command allows a user to delete a hosting NE from a hosting NE set.

**NOTE 1:** The name is mandatory and must refer to an existing *HostingNESet* in the system.

**NOTE 2:** The *hostingNEs* part of the *hostingNEArray* must be part of the provided set.

### 5.112.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteHostingNEHostingNESet"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.HostingNESetData
        name="aHostingSet2"
        systemSet="">
        <hostingNEArray>
          <string value="testAS"/>
        </hostingNEArray>
      </com.broadsoft.protocols.nsoss.HostingNESetData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.112.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="deleteHostingNEHostingNESet"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.113 addTrunkGroup

### 5.113.1 Description

This command is used to add a new trunk group in the system.

**NOTE 1:** The name is mandatory and must be unique.

**NOTE 2:** The *groupId* is mandatory and must refer to a valid *groupId* in the system.

**NOTE 3:** The *hostNE* is mandatory and must refer to an existing *HostingNE* in the system.

**NOTE 4:** The *groupId* must be assigned to the provided *hostNE*.

### 5.113.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="addTrunkGroup"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.TrunkGroupData
        name="aTrunkGroup"
        groupId="Melb3000"
        hostNE="ASaus"
        newName=""
        identity=""
        otgDtgIdentity="">
      </com.broadsoft.protocols.nsooss.TrunkGroupData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.113.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
  version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="addTrunkGroup"
  elementStartIndex="-1"
  loginId="admin"
  numberOfElements="-1"
  result="0"
  transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
```

```
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.114 deleteTrunkGroup

### 5.114.1 Description

This command is used to delete an existing trunk group from the system.

**NOTE 1:** The name is mandatory and must be unique.

**NOTE 2:** The *groupId* is mandatory and must refer to a valid *groupId* in the system.

**NOTE 3:** The *hostNE* is mandatory and must refer to an existing *HostingNE* in the system.

**NOTE 4:** The *groupId* must be assigned to the provided *hostNE*.

### 5.114.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
commandType="deleteTrunkGroup"
loginId="admin"
transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsooss.TrunkGroupData
name="aTrunkGroup"
groupId="Melb3000"
hostNE="ASaus"
newName=""
identity=""
otgDtgIdentity="">
  </com.broadsoft.protocols.nsooss.TrunkGroupData>
  </commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.114.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="deleteTrunkGroup"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
```

```
</com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.115 modifyTrunkGroup

### 5.115.1 Description

This command is used to modify *TrunkGroup*-related attributes in the system.

**NOTE 1:** The name is mandatory and must be unique.

**NOTE 2:** The *groupId* is mandatory and must refer to a valid *groupId* in the system.

**NOTE 3:** The *hostNE* is mandatory and must refer to an existing *HostingNE* in the system.

**NOTE 4:** The *groupId* must be assigned to the provided *hostNE*.

### 5.115.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="modifyTrunkGroup"  
    loginId="admin"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsoss.TrunkGroupData  
        name="aTrunkGroup"  
        groupId="Melb3000"  
        hostNE="ASaus"  
        newName=""  
        identity="anIdentity@identity.id"  
        otgDtgIdentity="">  
      </com.broadsoft.protocols.nsoss.TrunkGroupData>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.115.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0"  
>  
<commandArray>  
<com.broadsoft.protocols.nsoss.Command commandType="modifyTrunkGroup"  
elementStartIndex="-1"  
loginId="admin"  
numberOfElements="-1"  
result="0"
```



```
transactionId="1"  
>  
<commandDataArray>  
</commandDataArray>  
</com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.116 getTrunkGroup

### 5.116.1 Description

This command is used to view a specific instance of a *TrunkGroup* from the system.

**NOTE 1:** The name is mandatory and must be unique.

**NOTE 2:** The *groupId* is mandatory and must refer to a valid *groupId* in the system.

**NOTE 3:** The *hostNE* is mandatory and must refer to an existing *HostingNE* in the system.

**NOTE 4:** The *groupId* must be assigned to the provided *hostNE*.

### 5.116.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsoss.Command  
    commandType="getTrunkGroup"  
    loginId="admin"  
    transactionId="1">  
    <commandDataArray>  
      <com.broadsoft.protocols.nsoss.TrunkGroupData  
        name="aTrunkGroup"  
        groupId="Melb3000"  
        hostNE="ASaus"  
        newName=""  
        identity=""  
        otgDtgIdentity="">  
      </com.broadsoft.protocols.nsoss.TrunkGroupData>  
    </commandDataArray>  
  </com.broadsoft.protocols.nsoss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.116.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"  
version="20.0"  
>  
<commandArray>  
<com.broadsoft.protocols.nsoss.Command commandType="getTrunkGroup"
```

```
    elementStartIndex="-1"
    loginId="admin"
    numberOfElements="-1"
    result="0"
    transactionId="1"
  >
<commandDataArray>
<com.broadsoft.protocols.nsoss.TrunkGroupData elementStartIndex="-1"
  groupId="Melb3000"
  hostNE="ASaus"
  identity="anidentity@identity.id"
  name="aTrunkGroup"
  newName=""
  numberOfElements="-1"
  otgDtgIdentity=""
  >
</com.broadsoft.protocols.nsoss.TrunkGroupData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.117 getTrunkGroups

### 5.117.1 Description

This command is used to view all *TrunkGroups*-related attributes in the system.

### 5.117.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
  version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getTrunkGroups"
    loginId="admin"
    transactionId="1">
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.117.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
  version="20.0"
  >
<commandArray>
<com.broadsoft.protocols.nsoss.Command commandType="getTrunkGroups"
  elementStartIndex="-1"
  loginId="admin"
  numberOfElements="-1"
  result="0"
  transactionId="1"
  >
<commandDataArray>
<com.broadsoft.protocols.nsoss.TrunkGroupData elementStartIndex="-1"
```

```
groupId="Melb3000"
hostNE="ASaus"
identity="anidentity@identity.id"
name="aTrunkGroup"
numberOfElements="-1"
>
</com.broadsoft.protocols.nsoss.TrunkGroupData>
<com.broadsoft.protocols.nsoss.TrunkGroupData elementStartIndex="-1"
groupId="Melb3000"
hostNE="ASaus"
name="test"
numberOfElements="-1"
>
</com.broadsoft.protocols.nsoss.TrunkGroupData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.118 addNEMaintenancePartition

### 5.118.1 Description

This command is used to add a new NE maintenance partition in the system.

**NOTE:** The name is mandatory and must be unique.

### 5.118.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEPartition1">
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.118.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="19.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="addNEMaintenancePartition"
elementStartIndex="-1"
```

```
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.119 modifyNEMaintenancePartition

### 5.119.1 Description

This command is used to modify a NE maintenance partition in the system.

**NOTE 1:** The name is mandatory and must be unique.

**NOTE 2:** The *systemPartition* is optional and specifies if the partition is a system partition.

### 5.119.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.NEMaintenancePartitionData
        name="NEPartition1"
        systemPartition="true">
      </com.broadsoft.protocols.nsooss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.119.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="19.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command
commandType="modifyNEMaintenancePartition"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
```

```
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.120 deleteNEMaintenancePartition

### 5.120.1 Description

This command is used to delete a NE maintenance partition in the system.

**NOTE:** The name is mandatory and must be unique.

### 5.120.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEPartition1">
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.120.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="deleteNEMaintenancePartition"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.121 getNEMaintenancePartition

### 5.121.1 Description

This command is used to get a NE maintenance partition from the system.

**NOTE:** The name is mandatory and must be unique.

### 5.121.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEPartition1">
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
```

### 5.121.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getNEMaintenancePartition"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEPartition1"
        systemPartition="true">
        <nodeArray>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSecho"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
          <com.broadsoft.protocols.nsoss.NodeData
            name="PSecho"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
        </nodeArray>
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.122 getNEMaintenancePartitions

### 5.122.1 Description

This command is used to get all NE maintenance partition from the system.

**NOTE:** The name is mandatory and must be unique.

### 5.122.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getNEMaintenancePartitions"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
```

### 5.122.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getNEMaintenancePartitions"
    loginId="admin"
    result="0"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEPartition1"
        systemPartition="true">
        <nodeArray>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSecho"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
          <com.broadsoft.protocols.nsoss.NodeData
            name="PSecho"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
        </nodeArray>
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEpartition2"
        systemPartition="false">
        <nodeArray>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSmtlEast"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
          <com.broadsoft.protocols.nsoss.NodeData
            name="PSmtlEast"
            nodeId="0">
```

```
        </com.broadsoft.protocols.nsoss.NodeData>
    </nodeArray>
    </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.123 addHostingNENodeNEMaintenancePartition

### 5.123.1 Description

This command is used to add a new Hosting NE node to a NE maintenance partition in the system.

**NOTE:** The *NEMaintenancePartitionData:name*, *NodeData:name*, and *nodeId* are mandatory.

### 5.123.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addHostingNENodeNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEpartition1">
        <nodeArray>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSecho"
            nodeId="1">
          </com.broadsoft.protocols.nsoss.NodeData>
        </nodeArray>
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.123.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="addHostingNENodeNEMaintenancePartition"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
```



```
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.124 deleteHostingNENodeNEMaintenancePartition

### 5.124.1 Description

This command is used to delete a Hosting NE node from a NE maintenance partition in the system.

**NOTE:** The *NEMaintenancePartitionData:name*, *NodeData:name*, and *nodeId* are mandatory.

### 5.124.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteHostingNENodeNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEpartition2">
        <nodeArray>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSmtlEast"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
          <com.broadsoft.protocols.nsoss.NodeData
            name="PSmtlEast"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
        </nodeArray>
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.124.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="deleteHostingNENodeNEMaintenancePartition"
elementStartIndex="-1"
loginId="admin"
```

```
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.125 addWSFNodeNEMaintenancePartition

### 5.125.1 Description

This command is used to add a new Web Server Farm node to a NE maintenance partition in the system.

**NOTE:** The *NEMaintenancePartitionData:name*, *NodeData:name*, and *nodeId* are mandatory.

### 5.125.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="addWSFNodeNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEpartition1">
        <nodeArray>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSPecho"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
        </nodeArray>
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.125.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="addWSFNodeNEMaintenancePartition"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
```

```
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.126 deleteWSFNodeNEMaintenancePartition

### 5.126.1 Description

This command is used to delete a Web Server Farm node from a NE maintenance partition in the system.

**NOTE:** The *NEMaintenancePartitionData:name*, *NodeData:name*, and *nodeId* are mandatory.

### 5.126.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="deleteWSFNodeNEMaintenancePartition"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.NEMaintenancePartitionData
        name="NEpartition2">
        <nodeArray>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSPmtlEast"
            nodeId="0">
          </com.broadsoft.protocols.nsoss.NodeData>
          <com.broadsoft.protocols.nsoss.NodeData
            name="XSPmtlEast "
            nodeId="1">
          </com.broadsoft.protocols.nsoss.NodeData>
        </nodeArray>
      </com.broadsoft.protocols.nsoss.NEMaintenancePartitionData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.126.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="20.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
commandType="deleteWSFNodeNEMaintenancePartition"
```

```
elementStartIndex="-1"  
loginId="admin"  
numberOfElements="-1"  
result="0"  
transactionId="1"  
>  
<commanddataArray>  
</commanddataArray>  
</com.broadsoft.protocols.nsooss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.127 getHostingNesFromGroupId

### 5.127.1 Description

This command is used to obtain the BroadWorks hosting network element(s) that is serving a given group Id.

**NOTE 1:** The *groupId* is mandatory.

**NOTE 2:** The *enterprise* is optional.

**NOTE 3:** More than one hostingNe and associated enterprise can be returned in the response. Additionally the same enterprise with two or more distinct serving hostingNe might be returned.

### 5.127.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="getHostingNesFromGroupId"  
    loginId="admin"  
    transactionId="0">  
    <commanddataArray>  
      <com.broadsoft.protocols.nsooss.GroupEnterpriseData  
        groupId="groupAlpha"  
        enterprise="enterpriseBeta">  
      </com.broadsoft.protocols.nsooss.GroupEnterpriseData>  
    </commanddataArray>  
  </com.broadsoft.protocols.nsooss.Command>  
</commandArray>  
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.127.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>  
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"  
version="20.0">  
<commandArray>  
  <com.broadsoft.protocols.nsooss.Command  
    commandType="getHostingNesFromGroupId"  
    loginId="admin"  
    result="0"
```

```
transactionId="0">
  <commandDataArray>
    <com.broadsoft.protocols.nsoss.GroupEnterpriseData
      groupId="groupAlpha"
      enterprise="enterpriseBeta">
      <groupHostingNeArray>
        <com.broadsoft.protocols.nsoss.GroupHostingNeData
          hostingNE="AS1"
          enterprise="enterpriseBeta">
        </com.broadsoft.protocols.nsoss.GroupHostingNeData>
      </groupHostingNeArray>
    </com.broadsoft.protocols.nsoss.GroupEnterpriseData>
  </commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.128 getEnterpriseTrunkDNRange

### 5.128.1 Description

This command is used to view a specific enterprise trunk number range instance in the system.

**NOTE:** The *numberRangeStart* is mandatory.

### 5.128.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="21.0">
<commandArray>
  <com.broadsoft.protocols.nsoss.Command
    commandType="getEnterpriseTrunkDNRange"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData
        numberRangeStart="12407260200">
      </com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData>
    </commandDataArray>
  </com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.128.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
  version="21.0"
  >
<commandArray>
<com.broadsoft.protocols.nsoss.Command
  commandType="getEnterpriseTrunkDNRange"
  elementStartIndex="-1"
```

```
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
<com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData
elementStartIndex="0"
enterprise="Broadsoft"
group="group1"
isOffNet="true"
numberOfElements="-1"
numberRangeEnd="12409880019"
numberRangeStart="12409880010"
numberRangeStartPrefix=""
servingAS="ASSsanity"
site="DFLT_SITE"
>
<dNRangeArray>
</dNRangeArray>
</com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.129 getEnterpriseTrunkDNRanges

### 5.129.1 Description

This command is used to view a list of enterprise trunk number ranges in the system with optional search criteria.

**NOTE:** The parameters of *numberRangeStartPrefix*, *servingAS*, *enterprise*, *site*, *group*, and *isOffNet* are optional.

### 5.129.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
version="21.0">
  <commandArray>
    <com.broadsoft.protocols.nsoss.Command
commandType="getEnterpriseTrunkDNRanges"
loginId="admin"
transactionId="1">
      <commandDataArray>
        <com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData
servingAS="ASSsanity"
isOffNet="true">
          </com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData>
        </commandDataArray>
      </com.broadsoft.protocols.nsoss.Command>
    </commandArray>
  </com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.129.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
  version="21.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command
commandType="getEnterpriseTrunkDNRRanges"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="4"
>
>
<commandDataArray>
<com.broadsoft.protocols.nsooss.EnterpriseTrunkDNRRangeData
elementStartIndex="0"
enterprise="Broadsoft"
groupId=""
isOffNet="true"
numberOfElements="-1"
numberRangeEnd="12409880019"
numberRangeStart="12409880010"
numberRangeStartPrefix=""
servingAS="ASsanity"
site="DFLT_SITE"
>
>
</com.broadsoft.protocols.nsooss.EnterpriseTrunkDNRRangeData>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.130 addEnterpriseTrunkDNRRanges

#### 5.130.1 Description

This command is used to add a list of enterprise trunk number ranges to the system.

**NOTE 1:** The parameters *servingAS*, *enterprise*, and *dNRRangeArray* are mandatory.

**NOTE 2:** The parameters *site*, *group*, and *isOffNet* are optional.

**NOTE 3:** Each new range is validated to guarantee the start and end numbers are valid E.164 numbers, have same length and country code, and the range does not overlap with other ranges in the system.

#### 5.130.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
  version="21.0">
  <commandArray>
    <com.broadsoft.protocols.nsooss.Command
      commandType="getEnterpriseTrunkDNRRanges"
      loginId="admin"
```

```
transactionId="1">
  <commandDataArray>
    <com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData
      servingAS="ASSanity"
      isOffNet="true">
    </com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData>
  </commandDataArray>
</com.broadsoft.protocols.nsoss.Command> </commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

### 5.130.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsoss.BroadsoftDocument protocol="NSOSS"
  version="21.0"
>
<commandArray>
<com.broadsoft.protocols.nsoss.Command
  commandType="getEnterpriseTrunkDNRanges"
  elementStartIndex="-1"
  loginId="admin"
  numberOfElements="-1"
  result="0"
  transactionId="4"
>
<commandDataArray>
<com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData
  elementStartIndex="0"
  enterprise="Broadsoft"
  groupId=""
  isOffNet="true"
  numberOfElements="-1"
  numberRangeEnd="12409880019"
  numberRangeStart="12409880010"
  numberRangeStartPrefix=""
  servingAS="ASSanity"
  site="DFLT_SITE"
>
</com.broadsoft.protocols.nsoss.EnterpriseTrunkDNRangeData>
</commandDataArray>
</com.broadsoft.protocols.nsoss.Command>
</commandArray>
</com.broadsoft.protocols.nsoss.BroadsoftDocument>
```

## 5.131 modifyEnterpriseTrunkDNRanges

### 5.131.1 Description

This command is used to modify the site, group, and off-net status of a list of enterprise trunk number ranges in the system.

**NOTE 1:** The parameters *servingAS*, *enterprise*, and *dNArray* are mandatory.

**NOTE 2:** The parameters *site*, *group*, and *isOffNet* are optional.



**NOTE 3:** Each modifying range is validated to be hosted by the specified hosting Network Element and enterprise.

### 5.131.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="21.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="modifyEnterpriseTrunkDNRanges"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseTrunkDNRangesData
        servingAS="ASsanity"
        enterprise="Broadsoft"
        isOffNet="false">
        <dNArray>
          <String value="12407260200"/>
          <String value="12407260300"/>
        </dNArray>
      </com.broadsoft.protocols.nsooss.EnterpriseTrunkDNRangesData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.131.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="21.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command
commandType="modifyEnterpriseTrunkDNRanges"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 5.132 deleteEnterpriseTrunkDNRanges

### 5.132.1 Description

This command is used to delete a list of enterprise trunk number ranges from the system.

**NOTE 1:** The parameters *servingAS*, *enterprise*, and *dNArray* are mandatory.

**NOTE 2:** Each range to be deleted is validated to be hosted by the specified hosting Network Element and enterprise.

### 5.132.2 Sample Input XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="21.0">
<commandArray>
  <com.broadsoft.protocols.nsooss.Command
    commandType="deleteEnterpriseTrunkDNRanges"
    loginId="admin"
    transactionId="1">
    <commandDataArray>
      <com.broadsoft.protocols.nsooss.EnterpriseTrunkDNRangesData
        servingAS="ASsanity"
        enterprise="Broadsoft">
        <dNArray>
          <String value="12407260200"/>
          <String value="12407260299"/>
        </dNArray>
      </com.broadsoft.protocols.nsooss.EnterpriseTrunkDNRangesData>
    </commandDataArray>
  </com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

### 5.132.3 Sample Output XML Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<com.broadsoft.protocols.nsooss.BroadsoftDocument protocol="NSOSS"
version="21.0"
>
<commandArray>
<com.broadsoft.protocols.nsooss.Command commandType="deleteEnterpriseTrunkDNRanges"
elementStartIndex="-1"
loginId="admin"
numberOfElements="-1"
result="0"
transactionId="1"
>
<commandDataArray>
</commandDataArray>
</com.broadsoft.protocols.nsooss.Command>
</commandArray>
</com.broadsoft.protocols.nsooss.BroadsoftDocument>
```

## 6 Validation Requirements

The OSS interface is designed as a machine-to-machine interface with validation done before the interface is activated. Data input through this interface that does not meet the validation requirements can have unpredictable results within the system, including the inability to make calls. The following table specifies the data requirements for fields used in this interface.

Element	Attribute	Maximum Length	Validation Rules
Command	transactionId	10	Must be a positive integer with a maximum value of "2147483647".
Enterprise	name	32	Characters <%>, <">, <'>, and <\> are not allowed in the string. Must begin with an alphanumeric character. Cannot contain a double colon <::>.
User Group/VoiceVPNRouting Data	groupId	32	Characters <%>, <">, <'>, and <\> are not allowed in the string. Must begin with an alphanumeric character. Cannot contain a double colon <::>.
Enterprise/User Group	intraLataCarrier, interLataCarrier, internationalCarrier	8	FK validation: Must be an existing carrier and the carrier must be provisioned for the requested level.
Enterprise/User Group/Hosting NE/Site	profile	12	FK validation: Must be an existing profile.
User Group/Alternate Address/VoiceVPNRoutingData	hostingNE	40	FK validation: Must be an existing hosting NE.
User Group/Enterprise Admin/Hosting NE/Private Policy Instance/Authorized Host/Site/EntNGWRoutingRoutingData/VoiceVPNRoutingDataData	enterprise	32	FK validation: Must be an existing enterprise.
Enterprise	description	80	Cannot begin with <">.
User Group	groupName	80	Characters <%>, <">, and <'> are not allowed in the string.
Enterprise Admin	userid	20	Should not contain characters other than letters, numbers, <_>, and <->. Must be at least two characters.
Enterprise Admin	password	20	Must contain alphanumeric characters only.
Enterprise Admin	firstName	30	Characters </>, <">, and <\> are not allowed in the string.
Enterprise Admin	lastName	30	Characters </>, <">, and <\> are not allowed in the string.

Element	Attribute	Maximum Length	Validation Rules
Hosting NE	name	40	Characters <%>, <">, <'>, and < > are not allowed in the string. Must begin with an alphanumeric character. Cannot contain a double colon <::>.
Hosting NE	countryCode	3	FK validation: Must be an existing country code in the system.
Hosting NE Resource NE Web Server Farm Node	state	-	<i>OffLine</i> or <i>OnLine</i>
Hosting NE Node	name	40	FK validation: Must be an existing hosting NE in the system.
Hosting NE Node	nodeld	1	Integer value in range (from 0 through 9).
Hosting NE Node	description	80	Cannot begin with <">.
Hosting NE Node	version	30	No validation.
Hosting NE Node	patchLevel	30	No validation.
Hosting NE Node	minimumPatchLevel	30	No validation.
Hosting NE Address	address	255	Must contain alphanumeric characters only, and <.> or <->.
Hosting NE Address/EntNGW RoutingRoutingData/ Resource NE/ Resource NE Address/Web Server Farm Address	cost	1	Integer value in range (from 0 through 9).
Hosting NE Address/EntNGWRoutin gRoutingData/ Resource NE/ Resource NE Address/Web Server Farm Address	weight	2	Integer value in range (from 0 through 99).
Hosting NE Address	port	5	Integer value in range (from 1024 through 65535). Special value of "-1" is also allowed, indicating that the Network Server does not return the port in the list of contacts.
Hosting NE Address	type	11	One of the following: Signaling, Access, DualRouting, or Alias.
Hosting NE Address	transport	11	One of the following: unspecified, tcp, or udp.
Hosting NE	type	10	One of the following: broadworks or other.

Element	Attribute	Maximum Length	Validation Rules
Hosting NE admin	userIdArray	4-161 (per userId)	Each userId must be a valid URL, with at most one <@> character.
Authorized Host/EntNGWRouting RoutingData	authHost	40	FK validation: Must be an existing hosting NE.
Profile	name	12	FK validation: Must be an existing profile.
DnData/ DnRangeData	dn/fromDn/toDn	16	Only characters from 0 to 9 and <+> are allowed. For ranges, lower and upper limit must be of equal length, and lower must be less than or equal to upper.
UrlData	url	138	Must be a valid URL, with at most one <@> character. The alias part (preceding the @) is up to 32 characters long, and the domain part is up to 105 characters long.
PrivatePolicyArray	string/value	20	FK validation: Must be an existing private routing policy.
Site	name	20	Characters <%>, <">, <'>, and <\> are not allowed in the string. Must begin with an alphanumeric character. It cannot contain a double colon <::>.
Site	description	40	Cannot begin with <">.
DnData/ DnRangeData/ UrlData/EntNGWRouting RoutingData HostingNeData	site  defaultSynchSite defaultRoutingSite	20	FK validation: Must be an existing site.
TreatmentData/VoiceVP NRoutingData/VoiceVPN PropertiesData	treatmentId	8	FK validation: Must be an existing treatment ID.
CallTypeData/EntNGWR outingRoutingData/EntN GWRoutingProperti esData	callType	4	FK validation: Must be an existing call type ID.
EntNGWRoutingRouting Data/VoiceVPNRoutingD ataData	outDM	80	Must comply with the Network Server digit manipulation standard syntax (for more information, see section <a href="#">7.1 Network Server Digit Manipulation Syntax</a> ).
EntNGWRoutingProperti esData/VoiceVPN RoutingData	description	40	No validation.
VoiceVPNRoutingData	locCode	15	Digits from 0 through 9. Prepending "+" also allowed.
VoiceVPNRoutingData	extMin, extMax	Integer [0, 100]	

Element	Attribute	Maximum Length	Validation Rules
VoiceVPNRoutingData	selector	-	Allowed values are: "PUBLIC", "PRIVATE", "ROUTE", and "TRMT".
VoiceVPNPropertiesData	description	30	No validation.
Resource NE	location	15	String composed of digits from 0 through 9, and 0 to 15 digits long. Must represent the most significant portion of an E.164 number.
Resource NE	name	40	Characters <%, <">, <'>, and <\> are not allowed in the string. Must begin with an alphanumeric character. It cannot contain a double colon <::>.
Resource NE	type	13	One of the following: Media Server, SIMPLE server, or UVS.
EnterpriseData UserGroupData LCADData	lcaid	32	String (0 to 32 characters).
EnterpriseData SiteData UserGroupData DefaultSourceId Data	sourceId	128	String (0 to 128 characters).
ResourceNeData	service	-	None, one, or more of the following: all, conf, ivr, repeater, fax, liveAudio. If more than one, format is: <service>,<service>,...
Web Server Farm	name	40	Characters <%, <">, <'>, and <\> are not allowed in the string. Must begin with an alphanumeric character. Cannot contain a double colon <::>.
Web Server Farm	description	80	No validation.
Web Server Farm Node	name	40	FK validation: Must be an existing Web Server Farm in the system.
Web Server Farm Node	nodeId	1	Integer value in range (from 0 through 9).
Web Server Farm Node	version	30	No validation.
Web Server Farm Node	patchLevel	30	No validation.
Web Server Farm Node	minimumPatchLevel	30	No validation.
Web Server Farm Node	description	80	Cannot begin with <">.
Web Server Farm Address	address	255	Must contain alphanumeric characters only and <. > or <->.
Web Server Farm Address	type	6	One of the following: Access or Alias.

Element	Attribute	Maximum Length	Validation Rules
NetworkElementMediaCodecData	neName	40	FK validation: Must be an existing network element name.
NetworkElementMediaCodecData	neType	-	{"hosting", "routing", "resource"}
NetworkElementMediaCodecData	codecBasedRtgSupported	-	{"false", "true"}
MediaCodecData	mediaCodecName	32	FK validation: Must be an existing mediaCodec name.
MediaCodecData	priority	2	Integer value in range (from 1 through 99).
SystemMediaCodecData	mediaCodec	32	FK validation: Must be an existing mediaCodec name.
EnterpriseData	sourceIdForEAPrefix	128	String (0 to 80 characters).
NetworkTranslationIndexData	nti	128	String (1 to 128 characters).

## 7 Appendix A: Network Server Digit Manipulation

### 7.1 Network Server Digit Manipulation Syntax

The Network Server implements a generic digit-manipulation mechanism that is used in the implementation of various parts of the system, such as in voiceVPN and EntNGWRouting dial plans. It defines common digit manipulations such as insertions, replacements, and partial deletions.

#### 7.1.1 Mode of Operation

The Network Server digit manipulation framework uses the following format rules:

- A digit manipulation can be composed of one or more operations. The maximum is eight.
- An operation performs a basic digit manipulation and is specified using a command.
- Commands are in an assembler-like format. Operations are specified by three letters. Operations can have zero or one “operand”, typed as a string or integer.
- Commands do not allow predicates or logic control.

The digit manipulation framework implements a concept of a cursor that an operator can move in a digit buffer to apply an operation. The cursor can be positioned before a digit or after a digit, but not on a digit. Position 0 is defined as just before the first digit.

#### 7.1.2 Supported Commands

The supported commands are described in the following table.

Command	Operand	Operation	Description
<i>Basic Commands</i>			
POS	Integer	Position	Sets the cursor position (as defined above) to the value of the given operand. If the operand is lower than zero, the whole positioning operation is invalid (rejected). If the operand is greater than the length of the digit sequence, the position is set to the length of the digit sequence, or after the last digit.
END	None	End	Sets the cursor position to the length of the digit sequence, or after the last digit.
MOV	Integer	Move	Changes the cursor position by the value of the operand. If the new position is lower than zero, it is immediately brought back to zero. If the new position is greater than the length of the digit sequence, the position is set to the length of the digit sequence, or after the last digit.
INS	String	Insert	Inserts the operand after the cursor position. The position stays unchanged (same value).



Command	Operand	Operation	Description
DEL	Integer	Delete	Deletes the number of digits specified by the operand after the cursor position. If there are fewer following digits than the operand indicates, all following digits are deleted. The position stays unchanged (same value).
LTR	String	Left Trim	Strips occurrences of any character included in the operand from the beginning (left) of the digit sequence.
RTR	String	Right Trim	Strips occurrences of any character included in the operand from the end (right) of the digit sequence.
<b><i>Extended Commands</i></b>			
OVW	String	Overwrite	Inserts in overwrite mode. Equivalent to DEL(n), followed by INS(opd), where n is the length of opd.
APP	String	Append	Appends the specified operand. Equivalent to END(), followed by INS(opd).
PRE	String	Prepend	Prepends the specified operand. Equivalent to POS(0), followed by INS(opd).
REP	String	Replace	Replaces the whole digit sequence with a new one. Position is reset to zero. Equivalent to DEL(n), followed by INS(opd), where n is an integer known to be larger than the length of the digit sequence.
TRM	Char	Trim	Equivalent to LTR(opd), followed by RTR(opd).

## 8 Appendix B: Network Server Errors

### 8.1 Network Server Error Codes

In some cases, the Network Server may not be able to complete an OSS request sent by a connected client. In such a case, the Network Server returns an XML response that contains an error code and a description. The following is a list of the most common error codes the Network Server may return when a failure occurs.

**NOTE:** The following error descriptions can be changed for an international market. The system provider can modify and translate these messages as required.

```
#####  
###  
# Substitutable Error Strings  
#  
# Anything between {*} is parameter substituted.  
#  
# NOTE: READ THIS HERE*****  
# In the strings, please do not include the " or '. This will cause the  
# web pages to have a javascript error.  
#  
#  
# Do NOT translate "choice" in the string {0,choice,0#bla|1#once|2#{0}  
times}  
# Note: in the above string, "bla" is just a placeholder, which will  
never be shown.  
#  
#####  
###  
  
#####  
# Tags prepended to the message id  
#####  
Error = Error  
Warning = Warning  
  
#####  
# digit manipulation operations  
#####  
APP = APPEND  
DEL = DELETE  
END = END  
INS = INSERT  
LTR = LEFT TRIM  
MOV = MOVE  
OVW = OVERWRITE  
POS = POSITION  
PRE = PREPEND  
REP = REPLACE ALL  
RTR = RIGHT TRIM  
TRM = TRIM  
  
#4000 = System Error  
#4001 = Unauthorized transaction.  
#4002 = Unknown transaction: {0}  
#4003 = Unauthorized request: {0}
```

```
#4004 = Unsupported request: {0}

#4006 = Missing required parameters: {0}

#4013 = The database is currently involved in upgrade or maintenance
activities. No configuration changes can be made at this time. Please
try again later or contact your system administrator.

#4018 = Wrong password for User ID.
#4022 = Database locked. Please try again later.

#####
# General validation
#####
10050 = Cannot resolve host address: {0}
10051 = Cost is out of range [{0},{1}].
10052 = Weight is out of range [{0},{1}].
10053 = Static cost is out of range [{0},{1}].
10054 = Static weight is out of range [{0},{1}].
10055 = Default cost is out of range [{0},{1}].
10056 = Default weight is out of range [{0},{1}].
10057 = Port is out of range [{0},{1}].
10058 = Badly formatted address: {0}
10059 = URL must consist of two parts delimited by the '@' character: {0}
10060 = Duplicate URL: {0}
10061 = Maximum length exceeded, data truncation occurred
10062 = The system must be restarted for this property to take effect
10063 = Please narrow your search, there are {0} matching entries.

#####
# Admin management errors
#####
10100 = User authorization must be Enterprise, Group Provisioning or
System.
10101 = Admin already exists: {0}.
10102 = Admin does not exist: {0}.
10103 = Can only delete admins of the same enterprise.
10104 = Administrators cannot delete themselves.
10105 = Can only get properties of admins in the same enterprise.
10106 = Old password is invalid.
10107 = Cannot change the access level for the last full access system
administrator.
10108 = UserId should not contain characters other than those allowed in
the location-part of an e-mail address (incl. @).
10109 = UserId must not be empty.
10110 = UserId must be at least {0} characters long.
10111 = UserId must not be longer than 50 characters.
10112 = UserId must not be null.
10113 = First name should not contain slashes, backslashes and double
quotes.
10114 = Last name should not contain slashes, backslashes and double
quotes.
10115 = First name should not be null.
10116 = Last name should not be null.
10117 = First name must not be longer than 50 characters.
10118 = Last name must not be longer than 50 characters.
10119 = Max login failures out of range (-1 to 9)
10120 = Min password length out of range (3 to 40)
10121 = Min password length should not be less than the sum of all the
required numbers, uppercase, lowercase and non-alphanumeric characters
which is {0}.
10122 = Invalid password. The password must be at least {0} characters;
```

```
10123 = - contain login ID
10124 = - contain old password
10125 = - be the reverse of the old password
10126 = including at least {0} number;
10127 = including at least {0} uppercase alpha char(s);
10128 = including at least {0} lowercase alpha char(s);
10129 = including at least {0} non-alphanumeric char(s);
10130 = In addition, it cannot:

#####
# Enterprise management errors
#####
10150 = Enterprise does not exist: {0}.
10151 = Carrier is referenced {0,choice,0#bla|1#once|2#{0} times} by
Enterprises and/or Groups.
10152 = Enterprise description should not start with a quote.
10153 = Enterprise description must not be longer than 80 characters.
10154 = NIL_ENTERPRISE is system-defined and cannot be added nor deleted.
10155 = Enterprise already exists: {0}.
10156 = Enterprise with user groups cannot be deleted.
10157 = Enterprise with sites cannot be deleted.
10158 = Enterprise with user admins cannot be deleted.
10159 = Enterprise is referenced in Hosting NEs and cannot be deleted.

#####
# VTR management errors
#####
10200 = Executing Enterprise Policies
10201 = No policies are assigned to this Enterprise
10202 = Post Enterprise Policies Processing
10203 = This call received a treatment and will not be completed
10204 = This call will be processed by the System Provider's policies
10205 = This call will not be processed by the Service Provider's public
policies
10206 = Executing Incoming Digit Manipulation
10207 = A digit manipulation is assigned to this group on this network
element
10208 = The dialed number may be modified
10209 = The modified dialed number is: {0}
10210 = The number is not a valid public number
10211 = The call will not be processed by other policies
10212 = Public DN Validation
10213 = Basic Configuration
10214 = Status: Disabled
10215 = Status: Enabled
10216 = Policy not triggered for PRIVATE Network Type
10217 = Policy not triggered for URL call
10218 = Policy not triggered for call type
10219 = NearEndDM
10220 = Applying DM using index: {0}
10221 = Performing screening between {0} and {1}
10222 = FarEndDM
10223 = Applying DM using index: {0}
10224 = Performing screening between {0} and {1}
10225 = Call Category: {0}; CIC added: {1}
10226 = No CIC change
10227 = Adding CIC to routing entry: cc={0}; callCategory={1}; cic={2}
10228 = No CIC added because this route is LOCAL
10229 = {0}DM: {1}
10230 = {0}DN: {1}
10231 = DM Engine
10232 = {0}
10233 = The Originating Number does not belong to this enterprise.
```

```
10234 = Hosting NE: {0} of originator has no signaling IP address.
10235 = Hosting NE: {0} is OffLine
10236 = Enterprise version of this policy already ran. Continue.
10237 = Policy not triggered for URL caller
10238 = Feature code dialing not supported
10239 = keepPFX present, prepending {0}
10240 = keepPFX present, ..no prefix

#####
# DIGMAN management errors
#####
10250 = DMI does not exist: {0}
10251 = DMI already exists: {0}
10252 = DMI calls inexistent DMI: {0}
10253 = Recursion not allowed between DMIs: {0}
#10254 = DMI referred by other DMIs: {0}
10255 = Index cannot be negative
10256 = Unknown command in {0}
10257 = Missing argument in {0}
10258 = Too many arguments in {0}
10259 = Missing opening ( in {0}
10260 = Missing closing ) in {0}
10261 = Extra token after parentheses in {0}
10262 = Empty Operation in {0}
10263 = Invalid Integer value in {0}
10264 = Negative Integers not allowed in {0}
10265 = Invalid digit {0} at index {1} in {2}
10266 = Character + not allowed in operand of this command in {0}
10267 = Character # not allowed in operand of this command in {0}
10268 = Character + allowed only as first character for this command in
{0}
10269 = Too many operations ({0}) in Digit Manipulation. Maximum is {1}.
10270 = Too many characters ({0}) in Digit Manipulation. Maximum is {1}.
10271 = Command {0} not allowed.
10272 = Invalid Digit Manipulation Index: {0}
10273 = Invalid key: {0}
10274 = DMI is referenced in the system.
10275 = DMI is referenced {0,choice,0#bla|1#once|2#{0} times} by DMIs.

#####
# User Group management errors
#####
#
# NOTE: User groups are called "groups" in the web
#
10300 = Group does not exist: {0}.
10302 = Group already exists: {0}.
10303 = Group does not belong to enterprise: {0}.
10304 = Enterprise admins cannot modify a group's enterprise.
10305 = Group name should not contain percentages, quotes and
apostrophes.
10306 = Group ID must start with a letter or digit.
10307 = Group ID must not contain percentages, quotes, apostrophes,
backslashes and the :: character sequence.
10308 = Group ID must not be empty.
10309 = Group ID must not be null.
10310 = Group ID is too long.
10311 = Group name must not be null.
10312 = Group name is too long.
10313 = Group name is not unique.
10314 = Hosting NE is referenced {0,choice,0#bla|1#once|2#{0} times} by
Groups.
10315 = Group {0} is referenced in the system.
```

```
#####
# DN/extension manipulation errors
#####
#
# Note: these are similar, but not identical
#       to digixlator errors
#
10350 = A directory number must be non-empty and consist of digits only.
10351 = An extension number must non-empty and consist of digits only.
10352 = FROM and TO must be of equal length for a directory number range.
10353 = FROM and TO must be of equal length for an extension number
range.
10354 = FROM must be less than TO for a directory number range.
10355 = FROM must be less than TO for an extension number range.
10356 = FROM directory number must be non-empty and consist of digits
only.
10357 = FROM extension number must non-empty and consist of digits only.
10358 = TO directory number must be non-empty and consist of digits only.
10359 = TO extension number must non-empty and consist of digits only.
10360 = A directory number may be at most {0} digits long.
10361 = An extension number may be at most {0} digits long.
10362 = FROM directory number may be at most {0} digits long.
10363 = FROM extension number may be at most {0} digits long.
10364 = TO directory number may be at most {0} digits long.
10365 = TO extension number may be at most {0} digits long.
10366 = E164 DN is not valid. It cannot be validated against the rules
defined in Country Codes (CC) and National Destination Codes (NDC). Check
DN: {0} [CC found= {1}; NDC found= {2}; DN length= {3}]
10367 = DN: {0} is used by group: {1}
10368 = Number management is disabled.
10369 = Disabling number management will delete all unassigned numbers in
the system.
10370 = DN does not exist: {0}
10371 = Extension already exists: {0}

#####
# Carrier manipulation errors
#####
#
#
10400 = Carrier name should start with a letter or digit and should not
contain percentages, quotes, double quotes, backslashes and the ::
character sequence.
10401 = Carrier name must not be null.
10402 = Carrier name must not be empty.
10403 = Carrier name is too long.
10404 = Carrier already exists: {0}
10405 = Carrier does not exist: {0}
10406 = Carrier: {0} is referenced in the system.
10407 = Carrier: {0} is defined by the system and cannot be deleted.
10408 = Entry for routing NE: {0} already exists.
10409 = Entry for routing NE: {0} does not exist.
10410 = Carrier Access Code (CAC) length is invalid.
10411 = The carrier level (Intra-LATA, Inter-LATA, or International) is
used in the system and cannot be changed.
10412 = Carrier {0} cannot have preferred routing NE entries.
10413 = Carrier Access Code {0} already exists.
10414 = Carrier Access Code cannot be modified.
10415 = Carrier {0} does not support the requested level: {1}
```

```
10416 = Carrier {0} does not support the requested level: {1}. All inter-
LATA calls will be blocked.
10417 = The carrier is used in the system. All inter-LATA calls using
this carrier will be blocked.

#####
# Routing NE manipulation errors
#####
#
#
10450 = Routing NE does not exist: {0}
10451 = Routing NE name should start with a letter or digit and should
not contain percentages, quotes, double quotes, backslashes and the ::
character sequence.
10452 = Routing NE name must not be null.
10453 = Routing NE name must not be empty.
10454 = Routing NE name is too long.
10455 = Routing NE already exists: {0}
10456 = Routing NE: {0} is referenced in the system.
10457 = Invalid or conflicting from/to range limits.
10458 = Routing entry starting with {0} does not exist.
10459 = Routing entry ending with {0} does not exist.
10460 = From/To range is invalid or associated values are not consistent
across the range.
10461 = Routing entry does not exist.
10462 = Address does not exist for this routing NE.
10463 = Cannot modify more than one routing entry at a time.
10464 = All associated routing entries, addresses, and mediaCodecs will
be deleted.
10465 = Address already exists for this routing NE.
10466 = No address to set.
10467 = DMI is referenced {0,choice,0#bla|1#once|2#{0} times} by routing
NE routing entries.
10468 = Routing entry already exists.
10469 = Routing entry is referenced.

#####
# Hosting NE manipulation errors
#####
#
#
10500 = Hosting NE: {0} does not exist.
10501 = Hosting NE name should start with a letter or digit and should
not contain percentages, quotes, double quotes, backslashes and the ::
character sequence.
10502 = Hosting NE name must not be null.
10503 = Hosting NE name must not be empty.
10504 = Hosting NE name is too long.
10505 = Hosting NE: {0} already exists.
10506 = Hosting NE: {0} is referenced in the system.
10507 = All nodes, addresses, and mediaCodecs for this hosting NE will be
deleted.
10508 = Node ID is out of range [0,9].
10509 = Unknown address type: {0}.
10510 = Hosting NE node already exists: {0} {1}.
10511 = Hosting NE node does not exist: {0} {1}.
10512 = Cannot delete default hosting NE node.
10513 = Hosting NE node: {0} {1} is referenced in the system.
10514 = Cannot set default flag to false: set another node as default
instead.
10515 = This address is already assigned to this hosting NE node.
10516 = No address to delete.
10517 = This address does not exist for this hosting NE node.
```

```
10518 = No address to set.
10519 = All associated addresses for this hosting NE node will be
deleted.
10520 = Invalid Hosting NE type.
10521 = Unknown transport type: {0}.

#####
# Media Server manipulation errors
#####
#
#
10550 = Resource NE: {0} does not exist.
10551 = Resource NE name should start with a letter or digit and should
not contain percentages, quotes, double quotes, backslashes and the ::
character sequence.
10552 = Resource NE name must not be null.
10553 = Resource NE name must not be empty.
10554 = Resource NE name is too long.
10555 = Resource NE: {0} already exists.
10556 = Entry for routing NE: {0} already exists.
10557 = This address is already assigned to this Resource NE.
10558 = Entry for routing NE: {0} does not exist.
10559 = This address is not assigned to this Resource NE.
10560 = Cannot modify more than one selection entry at a time.
10561 = All associated media server selection entries, addresses, and
mediaCodecs will be deleted.
10562 = Resource NE: {0} is referenced in the system.
10563 = Resource NE: {0} is not a Media Server.
10564 = Resource NE: {0} is not a SIMPLE Server.
10565 = No address to set.
10566 = All associated addresses and mediaCodecs will be deleted.
10567 = Services settings only apply to Media Servers.
10568 = Service {0} does not exist.
10569 = Service {0} is already mapped to Media Server {1}.
10570 = Service {0} is not applicable for Media Server {1}.

#####
# Country Code manipulation errors
#####
#
#
10600 = Country Code: {0} does not exist.
10601 = Carrier is referenced {0,choice,0#bla|1#once|2#{0} times} in
Country Codes.

#####
# Routing Profile manipulation errors
#####
#
#
10650 = Routing Profile: {0} does not exist.

#####
# Location code validation (for routing NEs, media servers)
#####
#
#
10700 = Location code will be empty.
10701 = Location code will be deleted.
10702 = North American locations should be in 1NPANXX format.
```



```
10703 = {0} is not a valid entry the current NNACL file for country: 1
({1}).
10704 = Location has to begin with a valid country code.
10705 = This location code is parsed as: Country: {0} ({1}) , National
location: {2}.

#####
# Ping and trace device test tools
#####
#
#
10750 = The {0} operation may take few seconds. The test is going to be
performed on the following addresses.
10751 = Unknown HostingNe {0}
10752 = Unknown RoutingNe {0}
10753 = Unknown ResourceNe {0}
10754 = Unknown device type received
10755 = Invalid command

#####
# Site management errors
#####
#
#
10800 = Site {1} already exists in enterprise {0}.
10801 = Site {1} does not exist in enterprise {0}.
10802 = Default site cannot be deleted.
10803 = Site ID must not be empty.
10804 = Site ID must not be null.
10805 = Site ID is too long.
10806 = Site ID must start with a letter or digit.
10807 = Site ID must not contain percentages, quotes, apostrophes,
backslashes and the :: character sequence.
10808 = Site {1} in enterprise {0} is referenced in the system.
10809 = Site is referenced {0,choice,0#bla|1#once|2#{0} times} by Hosting
NEs.
10810 = Site is referenced {0,choice,0#bla|1#once|2#{0} times} by DNSs.
10811 = Site is referenced {0,choice,0#bla|1#once|2#{0} times} by URLs.
10812 = Site is referenced {0,choice,0#bla|1#once|2#{0} times} by
Extensions.

#####
# PM Reporting
#####
10850 = Hostname already exists.
10851 = Hostname not found.

#####
# Intl Call Screening
#####
10900 = Local Calling Area ID {0} is referenced in the system.
10901 = LCA ID is referenced {0,choice,0#bla|1#once|2#{0} times} by
Enterprises and User Groups.
10902 = LCA ID is referenced {0,choice,0#bla|1#once|2#{0} times} by Local
Calling Areas.
10903 = LCA ID is referenced {0,choice,0#bla|1#once|2#{0} times} by
National Destination Codes.
10904 = LCA ID does not exist: {0}
10905 = LCA Id already exists in LCAs based on LATA/Rate Centers: {0}
10906 = LCA Id already exists in country code {0}: {1}
```

```
10911 = Rate Center {0}, in Zone {1}, in Country Code {2}, is referenced
in the system.
10912 = Rate Center is referenced {0,choice,0#bla|1#once|2#{0} times} by
Local Calling Areas.
10913 = Rate Center is referenced {0,choice,0#bla|1#once|2#{0} times} by
National Destination Codes.
10914 = Zone/Rate Center already exists in country code {0}: {1}/{2}
10915 = Zone/Rate Center does not exist in country code {0}: {1}/{2}

10921 = Zone {0}, in Country Code {1}, is referenced in the system.
10922 = Zone is referenced {0,choice,0#bla|1#once|2#{0} times} by Local
Calling Areas.
10923 = Zone already exists in country code {0}: {1}
10924 = Zone does not exist in country code {0}: {1}

#####
# Policy
#####
10931 = Unknown policy: {0}
10932 = {0} instance does not exist for: {1}
10933 = {0} instance already exists for: {1}
10934 = Unlicensed policy: {0}
10935 = {0} instance already exists
10936 = {0} instance does not exist
10937 = {0} instances do not exist

#####
# Profile
#####
10941 = Profile already exists: {0}
10942 = An instance of a policy already exists for this profile. The
instance needs to be deleted first.
10943 = Profile in use, delete not allowed for: {0}
10944 = Profile does not exist: {0}

#####
# Treatment
#####
10951 = Treatment Mapping already exists: {0}
10952 = Treatment Mapping does not exist: {0}
10953 = Trying to delete system defined treatment
10954 = Treatment Mapping used in the system

#####
# NDC
#####
10961 = NDC already exists: {0}
10962 = NDC does not exist: {0}
10963 = Character other than 0-9 found in NDC: {0}

#####
# Call types
#####
10971 = CallType already exists: {0}
10972 = CallType does not exist: {0}
10973 = Call type {0} still in use in the system
10974 = System defined call types cannot be deleted

#####
# Domains
#####
10981 = Domain already exists: {0}
10982 = Domain does not exist: {0}
```

```
10983 = Domain still in use in the system.

#####
# Urls
#####
10991 = Url already exists: {0}
10992 = Url does not exist: {0}
10993 = LoginId Url already exists: {0}
10994 = LoginId Url: {0} is in conflict with an adminId for hosting NE:
{1}

#####
# Web Server Farms
#####
16000 = All nodes and addresses for this Web Server Farm will be deleted.
16001 = Web Server Farm: {0} does not exist.
16002 = Web Server Farm: {0} already exists.
16003 = Web Server Farm: {0} is referenced in the system.
16004 = Web Server Farm node: {0} {1} is referenced in the system.
16005 = Web Server Farm node already exists: {0} {1}.
16006 = Web Server Farm name should start with a letter or digit and
should not contain percentages, quotes, double quotes, backslashes and
the :: character sequence.
16007 = Web Server Farm name must not be null.
16008 = Web Server Farm name must not be empty.
16009 = Web Server Farm name is too long.
16010 = All associated addresses for this Web Server Farm node will be
deleted.
16011 = Web Server Farm node does not exist: {0} {1}.
16012 = This address is already assigned to this Web Server Farm node.
16013 = Cannot delete the last node for this Web Server Farm: {0}
16014 = Address {0} does not exist for Web Server Farm node: {1} {2}.
16015 = Web Server Farm address type {0} not supported for address {1}.

#####
# OSS Generic
#####
16050 = Invalid value length for value {0} in element {1}. Allowed range
is [{2} to {3}].
16051 = {0} and {1} not specified in request. The response contains only
the first {2} entries.
16052 = {0} not specified in request. The response contains only the
first {1} entries.
16053 = {0} not specified in request. The response contains a maximum of
{1} entries.
16054 = {0} is too large. The response contains a maximum of {1}
entries.
16055 = Please narrow your search, there are {0} matching records.

#####
# Media Codec
#####
16100 = The media name {0} already exists in the system as a media name
or codec name.
16101 = The media name {0} cannot be deleted. It is referenced by at
least one Hosting NE.
16102 = The media name {0} cannot be deleted. It is referenced by at
least one Routing NE.
16103 = The media name {0} cannot be deleted. It is referenced by at
least one Resource NE.
16104 = The media name {0} cannot be deleted. It is still referenced in
the system by table {1}.
16105 = The media type {0} already exists in the system.
```

16106 = The media name {0} does not exist.  
16107 = The media name {0} cannot be deleted. It is referenced by at least one codec name.  
16108 = The codec type {0} with payload type {1} and media type {2} already exists in the system.  
16109 = The payload type {0} is already defined in the system.  
16110 = The Hosting NE {0} already supports mediaCodec {1}.  
16111 = Delete failed. The Hosting NE {0} does not currently support mediaCodec {1}.  
16112 = The codec type {0} with media type {1} already exists in the system.  
16113 = The codec name {0} already exists in the system as a media name or codec name.  
16114 = The codec name {0} cannot be deleted. It is referenced by at least one Hosting NE.  
16115 = The codec name {0} cannot be deleted. It is referenced by at least one Routing NE.  
16116 = The codec name {0} cannot be deleted. It is referenced by at least one Resource NE.  
16117 = The codec name {0} cannot be deleted. It is still referenced in the system by table {1}.  
16118 = The codec name {0} does not exist.  
16119 = The mediaCodec {0} does not exist.  
16120 = The Routing NE {0} already supports mediaCodec {1}.  
16121 = The Resource NE {0} already supports mediaCodec {1}.  
16122 = Delete failed. The Routing NE {0} does not currently support mediaCodec {1}.  
16123 = Delete failed. The Resource NE {0} does not currently support mediaCodec {1}.  
16124 = Invalid network element type: {0}  
16125 = The media name {0} should start with a letter or digit and should not contain percentages, commas, quotes, double quotes, backslashes and the :: character sequence.  
16126 = The codec name {0} should start with a letter or digit and should not contain percentages, commas, quotes, double quotes, backslashes and the :: character sequence.

## Index

- Access control, 25
- addAuthorizedHost, 123
- addDataCenter, 186
- addDefaultSourceId, 172
- addEnterpriseAdmin, 104
- addHostingNe, 82
- addHostingNeAddr, 96
- addHostingNEAdministrators, 101
- addHostingNEHostingNESet, 196
- addHostingNeNode, 90
- addHostingNeNodeNEMaintenancePartition, 208
- addHostingNESet, 191
- Adding
  - addDataCenter, 186
  - addHostingNEHostingNESet, 196
  - addHostingNeNodeNEMaintenancePartition, 208
  - addHostingNESet, 191
  - addNEMaintenancePartition, 203
  - addTrunkGroup, 198
  - addWSFNodeNEMaintenancePartition, 210
  - Administrator loginIds to hosting network element, 101
  - Default Source ID, 172
  - Enterprise policy data entries, 130
  - Enterprise policy instance data entries, 127
  - Line/Port, 175
  - mediaCodecs to existing list, 166
  - NetworkTranslationIndex, 179
  - Unassigned directory numbers, 143
- addLinePort, 175
- addNEMaintenancePartition, 203
- addNetworkElementMediaCodec, 166
- addNetworkTranslationIndex, 179
- addPrivatePolicyInstance, 62
- addPrivatePolicyInstanceDataEntry, 127
- addPrivatePolicyPropertiesDataEntry, 130
- addSite, 76
- addTrunkGroup, 198
- addUnassignedNumber, 143
- addWebServerFarm, 149
- addWebServerFarmAddr, 160
- addWebServerFarmNode, 154
- addWSFNodeNEMaintenancePartition, 210
- Aggregate OCI commands, 25
- assignAddresses, 111
- Assigning, directory numbers and/or URLs, 111
- Authentication, login, and logout, 53
  - requestAuthentication, 55
  - requestLogin, 55
  - requestLogout, 56
- Authorization levels, 25
- Authorizing, FarEndHopOff or EntNGWRouting policy instance, 123
- Changes from previous releases, 19
- Changing, attributes of existing directory numbers and/or URLs, 113
- Commands
  - Network Server, description, 224
  - Service provider
    - addAuthorizedHost, 123
    - addDataCenter, 186
    - addDefaultSourceId, 172
    - addEnterpriseAdmin, 104
    - addHostingNe, 82
    - addHostingNeAddr, 96
    - addHostingNEAdministrators, 101
    - addHostingNeNode, 90
    - addHostingNeNodeNEMaintenancePartition, 208
    - addHostingNESet, 191
    - addLinePort, 175
    - addNEMaintenancePartition, 203
    - addNetworkElementMediaCodec, 166
    - addNetworkTranslationIndex, 179
    - addPrivatePolicyInstance, 62
    - addSite, 76
    - addTrunkGroup, 198
    - addWebServerFarm, 149
    - addWebServerFarmAddr, 160
    - addWebServerFarmNode, 154
    - addWSFNodeNEMaintenancePartition, 210
  - assignAddresses, 111
  - deleteAuthorizedHost, 124
  - deleteDataCenter, 187
  - deleteDefaultSourceId, 173
  - deleteEnterprise, 72, 182
  - deleteEnterpriseAdmin, 105
  - deleteHostingNe, 84
  - deleteHostingNeAddr, 97
  - deleteHostingNEAdministrators, 102
  - deleteHostingNEHostingNESet, 197
  - deleteHostingNeNode, 91
  - deleteHostingNeNodeNEMaintenancePartition, 209
  - deleteHostingNESet, 192
  - deleteLinePort, 176
  - deleteNEMaintenancePartition, 205
  - deleteNetworkElementMediaCodec, 167
  - deleteNetworkTranslationIndex, 182
  - deletePrivatePolicyInstance, 63
  - deleteSite, 79
  - deleteTrunkGroup, 199
  - deleteUserGroup, 117
  - deleteWebServerFarm, 150
  - deleteWebServerFarmAddr, 161
  - deleteWebServerFarmNode, 155

deleteWSFNodeNEMaintenance Partition, 211  
getAllCarriers, 59  
getAllPrivatePolicies, 61  
getAllServiceProviderProfiles, 58  
getAllSystemMediaCodec, 164  
getAuthorizedHosts, 125  
getDataCenter, 189  
getDataCenters, 190  
getDefaultSourceId, 174  
getEnterprise, 73  
getEnterpriseAdmin, 108  
getEnterpriseAdmins, 107  
getEnterprises, 74  
getHostingNe, 89  
getHostingNeAddr, 100  
getHostingNEAdministrator, 103  
getHostingNeNode, 95  
getHostingNeNodes, 93  
getHostingNes, 87  
getHostingNESets, 195  
getHostingNesFromGroupID, 212, 213, 214, 215, 216, 218  
getHostsForEnterprise, 75  
getLinePorts, 178  
getLocalCallingAreas, 60  
getNEMaintenancePartition, 206  
getNEMaintenancePartitions, 207  
getNetworkElementMediaCodec, 170  
getNetworkTranslationIndex, 183  
getNetworkTranslationIndexes, 184  
getPeers, 185  
getPrivatePolicyInstance, 66  
getSite, 80  
getSites, 81  
getTrunkGroup, 201  
getTrunkGroups, 202  
getUserGroup, 118  
getUserGroups, 120  
getWebServerFarm, 152, 153  
getWebServerFarmAddr, 163  
getWebServerFarmNode, 159  
getWebServerFarmNodes, 157  
modifyAddresses, 113  
modifyDataCenter, 188  
modifyDefaultSourceId, 173  
modifyEnterprise, 70  
modifyEnterpriseAdmin, 106  
modifyHostingNe, 85  
modifyHostingNeAddr, 98  
modifyHostingNeNode, 92  
modifyHostingNESet, 193  
modifyLinePort, 177  
modifyNEMaintenancePartition, 204  
modifyNetworkElementMediaCodec, 169  
modifyNetworkTranslationIndex, 181  
modifyPrivatePolicyInstance, 65  
modifySite, 78  
modifyTrunkGroup, 200  
modifyUserGroup, 116  
modifyWebServerFarm, 151  
modifyWebServerFarmAddr, 162  
modifyWebServerFarmNode, 156  
synchronizeEnterprise, 68  
synchronizeUserGroup, 109  
unassignAddresses, 114

System provider  
addPrivatePolicyInstanceDataEntry, 127  
addPrivatePolicyPropertiesDataEntry, 130  
addUnassignedNumber, 143  
deletePrivatePolicyInstanceDataEntry, 140  
deletePrivatePolicyPropertiesDataEntry, 141  
deleteUnassignedNumber, 144  
getCallTypes, 126  
getPrivatePolicyInstanceDataEntry, 135  
getPrivatePolicyPropertiesDataEntry, 138  
getResourceNeAddr, 148  
getResourceNes, 146  
getTreatments, 126  
getUnassignedNumber, 145  
modifyPrivatePolicyInstanceDataEntry, 131  
modifyPrivatePolicyPropertiesDataEntry, 134

System provisioning  
Open Client Interface-Provisioning, 27  
Operations Support System (OSS), 29  
OSS commands, 58

Creating  
Enterprise, 68  
Enterprise site, 76  
New enterprise administrator, 104  
New hosting network element, 82  
New private policy instance, 62  
New Web Server Farm, 149  
User group, 109

Defining  
New address for  
Existing hosting network element node, 96  
Existing Web Server Farm node, 160  
New node for  
Existing hosting network element, 90  
Existing Web Server Farm, 154

deleteAuthorizedHost, 124  
deleteDataCenter, 187  
deleteDefaultSourceId, 173  
deleteEnterprise, 72  
deleteEnterpriseAdmin, 105  
deleteHostingNe, 84  
deleteHostingNeAddr, 97  
deleteHostingNEAdministrators, 102  
deleteHostingNEHostingNESet, 197  
deleteHostingNeNode, 91  
deleteHostingNENodeNEMaintenancePartition, 209  
deleteHostingNESet, 192  
deleteLinePort, 176  
deleteNEMaintenancePartition, 205  
deleteNetworkElementMediaCodec, 167  
deleteNetworkTranslationIndex, 182  
deletePrivatePolicyInstance, 63

deletePrivatePolicyInstanceDataEntry, 140  
deletePrivatePolicyPropertiesDataEntry, 141  
deleteSite, 79  
deleteTrunkGroup, 199  
deleteUnassignedNumber, 144  
deleteUserGroup, 117  
deleteWebServerFarm, 150  
deleteWebServerFarmAddr, 161  
deleteWebServerFarmNode, 155  
deleteWSFNodeNEMaintenancePartition, 211  
Deleting  
    Administrator loginIds from hosting network element, 102  
    Default Source ID, 173  
    deleteDataCenter, 187  
    deleteHostingNEHostingNESet, 197  
    deleteHostingNENodeNEMaintenancePartition, 209  
    deleteHostingNESet, 192  
    deleteNEMaintenancePartition, 205  
    deleteTrunkGroup, 199  
    Enterprise policy data entries, 141  
    Enterprise policy instance data entries, 140  
    Existing enterprise, 72  
    Existing enterprise administrator, 105  
    Existing hosting network element, 84, 91  
    Existing hosting network element node address, 97  
    Existing site, 79  
    Existing Web Server Farm, 150  
    Existing Web Server Farm node, 155  
    Existing Web Server Farm node address, 161  
    Line/Port, 176  
    mediaCodecs from existing list, 167  
    NetworkTranslationIndex, 182  
    Private policy instance, 63  
    Unassigned directory numbers, 144  
    User groups, 117  
Digit manipulation, 224  
    Commands, 224  
    Mode of operation, 224  
    Syntax, 224  
DTD, 29  
Element description, 42  
Enterprises, 74  
Error codes, Network Server, 226  
External provisioning  
    Open Client Interface-Provisioning (OCI-P), 24  
    Operations Support System (OSS), 24  
Get command, 51  
    Narrowing results returned, 51  
getAllCarriers, 59  
getAllPrivatePolicies, 61  
getAllServiceProviderProfiles, 58  
getAllSystemMediaCodec, 164  
getAuthorizedHosts, 125  
getCallTypes, 126  
getDataCenter, 189  
getDataCenters, 190  
getDefaultSourceIds, 174  
getEnterprise, 73  
getEnterpriseAdmin, 108  
getEnterpriseAdmins, 107  
getHostingNe, 89  
getHostingNeAddr, 100  
getHostingNEAdministrator, 103  
getHostingNeNode, 95  
getHostingNeNodes, 93  
getHostingNes, 87  
getHostingNESet, 194  
getHostingNESets, 195  
getHostingNesFromGroupId, 212, 213, 214, 215, 216, 218  
getLinePorts, 178  
getLocalCallingAreas, 60  
getNEMaintenancePartition, 206  
getNEMaintenancePartitions, 207  
getNetworkElementMediaCodec, 170  
getNetworkTranslationIndex, 183  
getNetworkTranslationIndexes, 184  
getPeers, 185  
getPrivate PolicyInstance, 66  
getPrivatePolicyInstanceDataEntry, 135  
getPrivatePolicyPropertiesDataEntry, 138  
getResourceNeAddr, 148  
getResourceNes, 146  
getSite, 80  
getSites, 81  
Getting  
    getDataCenters, 190  
    getPeers, 185  
    Line/Ports, 178  
getTreatments, 126  
getTrunkGroup, 201  
getTrunkGroups, 202  
getUnassignedNumber, 145  
getUserGroup, 118  
getUserGroups, 120  
getWebServerFarm, 152, 153  
getWebServerFarmAddr, 163  
getWebServerFarmNode, 159  
getWebServerFarmNodes, 157  
modifyAddresses, 113  
modifyDataCenter, 188  
modifyDefaultSourceId, 173  
modifyEnterprise, 70  
modifyEnterpriseAdmin, 106  
modifyHostingNe, 85  
modifyHostingNeAddr, 98  
modifyHostingNeNode, 92  
modifyHostingNESet, 193  
Modifying

- Attributes of private policy instance, 65
- Default Source ID, 173
- Enterprise policy data entries, 134
- Enterprise policy instance data entries, 131
- Existing enterprise, 70
- Existing hosting network element node, 92
- Existing hosting network element node address, 98
- Existing site, 78
- Existing user groups, 116
- Existing Web Server Farm node, 156
- Existing Web Server Farm node address, 162
- Line/Port, 177
- modifyDataCenter, 188
- modifyHostingNESet, 193
- modifyNEMaintenancePartition, 204
- modifyTrunkGroup, 200
- NetworkTranslationIndex, 181
- Properties of
  - Existing enterprise administrator, 106
  - Existing hosting network element, 85
  - Existing Web Server Farm, 151
- modifyLinePort, 177
- modifyNEMaintenancePartition, 204
- modifyNetworkElementMediaCodec, 169
- modifyNetworkTranslationIndex, 181
- modifyPrivatePolicyInstance, 65
- modifyPrivatePolicyInstanceDataEntry, 131
- modifyPrivatePolicyPropertiesDataEntry, 134
- modifySite, 78
- modifyTrunkGroup, 200
- modifyUserGroup, 116
- modifyWebServerFarm, 151
- ModifyWebServerFarmAddr, 162
- modifyWebServerFarmNode, 156
- Network Server
  - Digit manipulation, 224
  - Error codes, 226
- OCI-P Interface, 24
- OCI-P XML Definition, 27
- OSS Interface, 24
- OSS XML Definition, 29
- Querying
  - All user groups, 120
  - Existing user groups, 118
  - Network Server for existing call types, 126
  - Network Server for existing enterprise, 73, 74
  - Network Server for existing site, 80, 81
  - Network Server for existing treatments, 126
  - Network Server for mediaCodecs, 164
  - Unassigned directory numbers, 145
- requestAuthentication, 55
- requestLogin, 55
- requestLogout, 56
- Requirements, validation, 219
- Resetting, list of mediaCodecs, 169
- Retrieving
  - Addresses defined for
    - Existing hosting network element node, 100
    - Existing Web Server Farm node, 163
  - Addresses for resource network element, 148
  - Attributes for private policy instance, 66
  - Default Source ID, 174
  - Enterprise policy data entries, 138
  - Enterprise policy instance data entries, 135
  - Existing hosting network element node, 93
  - Existing Web Server Farm node, 157
  - getDataCenter, 189
  - getHostingNESet, 194
  - getHostingNESets, 195
  - getHostingNesFromGroupId, 212, 213, 214, 215, 216, 218
  - getNEMaintenancePartition, 206
  - getNEMaintenancePartitions, 207
  - getTrunkGroup, 201
  - getTrunkGroups, 202
  - List of administrator loginIds, 103
  - List of authorized hosts, 125
  - List of carriers, 59
  - List of local calling areas, 60
  - List of mediaCodecs, 170
  - List of private routing policies, 61
  - List of public routing profiles, 58
  - NetworkTranslationIndex, 183
  - NetworkTranslationIndexes, 184
  - Properties of
    - Existing enterprise administrator, 107, 108
    - Existing hosting network element, 87, 89
    - Existing hosting network element node, 95
    - Existing Web Server Farm, 152, 153
    - Existing Web Server Farm node, 159
  - Resource network elements, 146
- synchronizeEnterprise, 68
- synchronizeUserGroup, 109
- unassignAddresses, 114
- Unassigning, directory numbers and/or URLs, 114
- Un-authorizing, FarEndHopOff or EntNGWRouting policy instance, 124
- Validation requirements, 219
- Validation, OCI, 26
- XML definition
  - OCI-P, 27
  - OSS, 29