IBM TotalStorage SAN Volume Controller

CIM Agent Developer's Reference

Version 1.2.1

SC26-7545-02

IBM TotalStorage SAN Volume Controller

CIM Agent Developer's Reference

Version 1.2.1

SC26-7545-02

Second Edition (October 2004)

Note: Before using this information and the product it supports, read the information in ″Notices.″

© Copyright International Business Machines Corporation 2003, 2004. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures ix

Tables xi

About this guide xvii
Who should use this guide? xvii
Summary of changes xvii
Summary of changes for GC26-7545-02 SAN Volume Controller CIM Agent Developer's Reference Release 1.2.1 xvii
Emphasis xviii
SAN Volume Controller library and related publications xviii
Related Web sites xx
How to order IBM publications xx
How to send your comments xxi

Chapter 1. Introduction 1
Storage Management Initiative Specification 1
Common Information Model 2
Common Information Model concepts 2
CIM agent 3
SAN Volume Controller 5
Common Information Model Agent for the SAN Volume Controller 8
Validating the truststore certificate expiration 9
Functional views of the Common Information Model Agent 10
Profile Overview 10
Physical Package 11
Server Profile 12
Extent mapping subprofile 13
ExtraCapacitySet subprofile 14
Access point subprofile 15
Cluster subprofile 16
Vendor-specific storage configuration operations 17
LUN masking 18
LUN creation subprofile 19
Copy services 20
Vendor-specific service mode subprofile 21
Vendor-specific cluster operations 22
Security service 23
Pool Manipulation 24

Chapter 2. Performing storage configuration 27
Storage configuration 27
Performing basic storage configuration 27
Adding a candidate node to a cluster 27
Creating a new storage pool 28
Modifying a storage pool 28
Creating a new storage volume 29

Chapter 3. Performing Copy Services 31
Copy Services 31
Creating a new FlashCopy relationship between storage volumes 31
Creating a FlashCopy relationship for a synchronized set 32
Creating a synchronous copy relationship between volumes in the same cluster 33

© Copyright IBM Corp. 2003, 2004

Creating a synchronous copy relationship between volumes in different clusters 33
Flash Copy state diagram 34
Sync Copy state diagram 35

Chapter 4. Performing LUN masking 37
LUN masking 37
Performing LUN masking 37

Chapter 5. Network Considerations 39
SLP based discovery 39
RemoteServiceAccessPoint 39

Chapter 6. CIM Agent object classes 41
Core object classes 41
IBMTSSVC_BackendController 41
IBMTSSVC_BackendVolume 48
IBMTSSVC_CandidateCluster 59
IBMTSSVC_CandidateNode 60
IBMTSSVC_CandidateStorageHardwareID. 66
IBMTSSVC_CandidateVolume 68
IBMTSSVC_Chassis 69
IBMTSSVC_Cluster 77
IBMTSSVC_Controller 84
IBMTSSVC_ControllerConfigurationService 91
IBMTSSVC_ControllerMaskingCapabilities 94
IBMTSSVC_Dumps 97
IBMTSSVC_FCPort 98
IBMTSSVC_Features 108
IBMTSSVC_FlashCopySynchronizedSet 109
IBMTSSVC_HardwareIdCollection 110
IBMTSSVC_IOGroup 112
IBMTSSVC_IOGroupSet 118
IBMTSSVC_Job 119
IBMTSSVC_MessageLog 124
IBMTSSVC_Node 132
IBMTSSVC_NodeVPD 137
IBMTSSVC_PrimordialStoragePool 138
IBMTSSVC_Privilege 140
IBMTSSVC_Product 142
IBMTSSVC_Provider 143
IBMTSSVC_RegisteredProfile 146
IBMTSSVC_RegisteredSubProfile 149
IBMTSSVC_RemoteCluster 152
IBMTSSVC_RemoteServiceAccessPoint 154
IBMTSSVC_RemoteVolume 160
IBMTSSVC_StorageCapabilities 161
IBMTSSVC_StorageConfigurationCapabilities 164
IBMTSSVC_StorageHardwareID 168
IBMTSSVC_StoragePool 169
IBMTSSVC_StorageSetting 171
IBMTSSVC_StorageVolume 175
IBMTSSVC_SyncCopySynchronizedSet 189
Service object classes 191
IBMTSSVC_ClusteringService 191
IBMTSSVC_PrivilegeManagementService 195
IBMTSSVC_StorageConfigurationService 198

IBMTSSVC_StorageHardwareIDManagementService 202
Security object classes 205
IBMTS_Account 206
IBMTS_AccountManagementService 208
IBMTS_CIMXMLCommunicationMechanism 212
IBMTS_IndicationFilter 217
IBMTS_NameSpace 218
IBMTS_ObjectManager 220
IBMTS_RegisteredProfile. 224
IBMTS_System 225
Association object classes 228
IBMTSSVC_AllocatedFromStoragePool 228
IBMTSSVC_AuthorizedCollection. 229
IBMTSSVC_AuthorizedStorageHardwareID 229
IBMTSSVC_AuthorizedSubject 230
IBMTSSVC_AuthorizedTarget 230
IBMTSSVC_AvailableHardwareID 230
IBMTSSVC_BackendControllerForVolume 231
IBMTSSVC_BasedOn 231
IBMTSSVC_ClusterController 232
IBMTSSVC_ClusterDumps 232
IBMTSSVC_ClusteringCandidate 233
IBMTSSVC_ClusteringServiceForSystem 233
IBMTSSVC_ClusterMaskingCapabilities 234
IBMTSSVC_ClusterPort 234
IBMTSSVC_ClusterScopeCandidateVolume 234
IBMTSSVC_ClusterScopeChassis 235
IBMTSSVC_ClusterScopeFCSet 235
IBMTSSVC_ClusterScopeIOGroup 235
IBMTSSVC_ClusterScopeNodeVPD 236
IBMTSSVC_ClusterScopePrivilege 236
IBMTSSVC_ClusterScopeProduct 236
IBMTSSVC_ClusterScopeSCSet 236
IBMTSSVC_ClusterVolume 237
IBMTSSVC_ComponentCS 237
IBMTSSVC_ComputerSystemPackage. 237
IBMTSSVC_ConnectedBackendController 238
IBMTSSVC_ControllerConfigurationServiceForSystem 238
IBMTSSVC_ControllerConfServiceMaskingCapabilities 239
IBMTSSVC_CopyCandidate 239
IBMTSSVC_ElementConformsToProfile 239
IBMTSSVC_FlashCopyStorageSynchronized 240
IBMTSSVC_FlashCopySynchronizedMember 242
IBMTSSVC_HardwareIDOnSystem 242
IBMTSSVC_HostedAccessPoint 243
IBMTSSVC_HostedJob 243
IBMTSSVC_HostedPrimordialPool 243
IBMTSSVC_HostedStoragePool 244
IBMTSSVC_HwIDCollectionOnSystem 244
IBMTSSVC_IndicationFiltersConformsToProfile. 244
IBMTSSVC_IndicationFiltersConformsToSubProfile 245
IBMTSSVC_IOGroupIdentity 245
IBMTSSVC_IOGroupPort 245
IBMTSSVC_ManagesCollection 246
IBMTSSVC_ManagesController 246
IBMTSSVC_ManagesHardwareID 246

IBMTSSVC_ManagesPrivilege 247
IBMTSSVC_MemberOfCollection 247
IBMTSSVC_MemberOfIOGroup 247
IBMTSSVC_NodeDumps 248
IBMTSSVC_PartnershipCandidate 249
IBMTSSVC_PoolCapabilities 249
IBMTSSVC_PrimordialPoolCapabilities 249
IBMTSSVC_PrimordialPoolComponent 250
IBMTSSVC_PrimordialPoolForController 250
IBMTSSVC_PrivilegeServiceForSystem 250
IBMTSSVC_ProductPhysicalComponent 251
IBMTSSVC_ProtocolControllerForPort 251
IBMTSSVC_ProtocolControllerForUnit 252
IBMTSSVC_ProviderInObjectManager 253
IBMTSSVC_RemotePartnership 254
IBMTSSVC_RemoteSystemVolume 254
IBMTSSVC_RequiresProfile 254
IBMTSSVC_SAPAvailableForElement 255
IBMTSSVC_StorageConfigurationServiceCapabilities 255
IBMTSSVC_StorageConfigurationServiceForSystem 255
IBMTSSVC_StorageHardwareIDManagementServiceForSystem 256
IBMTSSVC_StoragePoolComponent 256
IBMTSSVC_SyncCopyStorageSynchronized 256
IBMTSSVC_SyncCopySynchronizedMember 259
IBMTSSVC_SystemBackendVolume 260
IBMTSSVC_SystemCandidateVolume 260
IBMTSSVC_SystemController 260
IBMTSSVC_SystemFCPort 261
IBMTSSVC_SystemFeatures 261
IBMTSSVC_SystemVolume 261
IBMTSSVC_SystemVPD 262
IBMTSSVC_UseOfMessageLog 263
IBMTSSVC_VolumeSettingData 263
IBMTSSVC_AccountManagementServiceForSystem 264
IBMTS_AccountOnCIMOM 265
IBMTS_AccountOnSystem 265
IBMTS_CommMechanismForManager 266
IBMTS_ElementConformsToProfile 266
IBMTS_HostedAccessPoint 266
IBMTS_HostedService 267
IBMTS_IndicationFiltersConformsToProfile 267
IBMTS_ManagesAccount 268
IBMTS_NamespaceInManager 268

Chapter 7. CIM Agent methods 269
Intrinsic methods 269
Associators() 269
AssociatorNames() 270
CreateInstance() 271
DeleteInstance() 272
EnumerateClasses() 272
EnumerateClassNames() 273
EnumerateInstances() 273
EnumerateInstanceNames() 274
ExecQuery() 275
GetClass() 275

GetInstance() . . . 276
GetProperty() . . . 276
ModifyInstance() . . . 277
References() . . . 277
ReferenceNames() . . . 278
SetProperty() . . . 279
Extrinsic methods . . . 279
Add2062Cluster() . . . 282
Add2145Cluster() . . . 283
AddHardwareIDsToCollection() . . . 283
AddNode() . . . 284
AssignAccess() . . . 285
AttachDevice() . . . 286
AttachReplica() . . . 287
BackupConfiguration() . . . 288
CancelIteration() . . . 289
Clean() . . . 289
ClearLog() . . . 290
Create2062Cluster() . . . 290
CreateHardwareIDCollection() . . . 291
CreateOrModifyStoragePool() . . . 292
CreateOrModifyElementFromStoragePool() . . . 295
CreateProtocolControllerWithPorts() . . . 297
CreateRemoteClusterPartnership() . . . 298
CreateReplica() . . . 299
CreateSetting() . . . 299
CreateStorageHardwareID() . . . 300
CreateSynchronizedSet() . . . 301
DeleteConfigurationBackup() . . . 302
DeleteHardwareIDCollection() . . . 302
DeleteProtocolController() . . . 303
DeleteRecord() . . . 304
DeleteRemoteClusterPartnership() . . . 305
DeleteStorageHardwareID() . . . 305
DeleteSynchronizedSet() . . . 306
DeleteStoragePool() . . . 307
DetachDevice() . . . 307
Dump() . . . 308
Enter() . . . 309
EvictNode() . . . 309
Exit() . . . 310
FixRecord() . . . 310
GetAllRecords() . . . 311
GetDump() . . . 311
GetFreeExtents() . . . 312
GetRecord() . . . 313
GetResetPasswordChangeFeatureStatus() . . . 313
GetSupportedSizeRange() . . . 314
GetSupportedSizes() . . . 314
IncludeBackendVolume() . . . 315
ListConfigurationBackups() . . . 316
MigrateVolume() . . . 316
ModifyErrorSettings() . . . 317
ModifyIPAddress() . . . 317
ModifyResetPasswordChangeFeature() . . . 318
ModifySynchronization() . . . 318

ModifySynchronizedSet() 320
PositionAtRecord() 322
PositionToFirstRecord() 323
PositionToFirstRecordRoot() 324
PositionToFirstRecordType() 324
Reload2062Node() 325
RemoveAccess() 326
RemoveCluster() 326
RequestDiscovery() 327
Reset2062Node() 327
RestoreConfiguration() 328
ReturnToStoragePool() 329
SetIOGroup() 330
SetLocale() 330
SetPasswords() 331
SetQuorum() 331
SetTimeZone() 332
StartStatisticsCollection() 332
StopStatisticsCollection() 333
Shutdown() 333
UnfixRecord() 334
Upgrade() 334
WriteRecord() 335

Accessibility 337

Notices 339
Trademarks. 340

Glossary 341

Index 345

Figures

1. A typical CIM Agent at work 4
2. A SAN Volume Controller node 6
3. Example of a SAN Volume Controller in a fabric 7
4. High-level overview of the CIM Agent for the SAN Volume Controller. 11
5. High-level overview of the physical package of the CIM Agent for the SAN Volume Controller. 12
6. High-level overview of the server profile of the CIM Agent for the SAN Volume Controller. 13
7. High-level overview of the extent mapping subprofile of the CIM Agent for the SAN Volume Controller. 14
8. High-level overview of the ExtraCapacitySet subprofile of the CIM Agent for the SAN Volume Controller. 15
9. High-level overview of the access point subprofile of the CIM Agent for the SAN Volume Controller. 16
10. Class diagram of Clustering instance 17
11. Class diagram of StorageConfiguration instance 18
12. Class diagram for LUN masking instances 19
13. High-level overview of the LUN creation subprofile of the CIM Agent for the SAN Volume Controller. 20
14. Class diagram of copy services instances. 21
15. High-level overview of the vendor-specific service mode subprofile of the CIM Agent for the SAN Volume Controller. 22
16. High-level overview of the vendor-specific cluster operations of the CIM Agent for the SAN Volume Controller. 23
17. Class diagram of security instances 24
18. High-level overview of pool manipulation of the CIM Agent for the SAN Volume Controller.. 25
19. Flash Copy state diagram of the CIM Agent for the SAN Volume Controller. 35
20. High-level overview of the Sync Copy state diagram of the CIM Agent for the SAN Volume Controller. 36

© Copyright IBM Corp. 2003, 2004

Tables

1. Emphasis descriptions xviii
2. Publications in the SAN Volume Controller library xix
3. Other IBM publications xx
4. Web sites xx
5. IBMTSSVC_BackendController properties 41
6. IBMTSSVC_BackendVolume properties 49
7. IBMTSSVC_CandidateCluster properties 60
8. IBMTSSVC_CandidateNode properties 60
9. IBMTSSVC_CandidateStorageHardwareID properties 67
10. IBMTSSVC_CandidateVolume properties 68
11. IBMTSSVC_Chassis properties 70
12. IBMTSSVC_Cluster properties 77
13. IBMTSSVC_Controller properties 84
14. IBMTSSVC_ControllerConfigurationService properties 91
15. IBMTSSVC_ControllerMaskingCapabilities properties 95
16. IBMTSSVC_Dumps properties 97
17. IBMTSSVC_FCPort properties 98
18. IBMTSSVC_Features properties 108
19. IBMTSSVC_FlashCopySynchronizedSet properties 109
20. IBMTSSVC_HardwareIdCollection properties 111
21. IBMTSSVC_IOGroup properties 112
22. IBMTSSVC_IOGroupSet properties 118
23. IBMTSSVC_Job properties 120
24. IBMTSSVC_MessageLog properties 125
25. IBMTSSVC_Node properties 132
26. IBMTSSVC_NodeVPD properties 138
27. IBMTSSVC_PrimordialStoragePool properties 138
28. IBMTSSVC_Privilege properties 141
29. IBMTSSVC_Product properties 142
30. IBMTSSVC_Provider properties 143
31. IBMTSSVC_RegisteredProfile properties 147
32. IBMTSSVC_RegisteredSubProfile properties 150
33. IBMTSSVC_RemoteCluster properties 153
34. IBMTSSVC_RemoteServiceAccessPoint properties 154
35. IBMTSSVC_RemoteVolume properties 160
36. IBMTSSVC_StorageCapabilities properties 162
37. IBMTSSVC_StorageConfigurationCapabilities properties 164
38. IBMTSSVC_StorageHardwareID properties 168
39. IBMTSSVC_StoragePool properties 169
40. IBMTSSVC_StorageSetting properties 172
41. IBMTSSVC_StorageVolume properties 175
42. IBMTSSVC_SyncCopySynchronizedSet properties 189
43. IBMTSSVC_ClusteringService properties 192
44. IBMTSSVC_PrivilegeManagementService properties 195
45. IBMTSSVC_StorageConfigurationService properties 199
46. IBMTSSVC_StorageHardwareIDManagementService properties 202
47. IBMTS_Account properties 206
48. IBMTS_AccountManagementService properties 208
49. IBMTS_CIMXMLCommunicationMechanism properties 212
50. IBMTS_IndicationFilter properties 218
51. IBMTS_NameSpace properties 218
52. IBMTS_ObjectManager properties 220
53. IBMTS_RegisteredProfile properties 224

© Copyright IBM Corp. 2003, 2004

54. IBMTS_System properties . . . 225
55. IBMTSSVC_AllocatedFromStoragePool references . . . 229
56. IBMTSSVC_AllocatedFromStoragePool properties . . . 229
57. IBMTSSVC_AuthorizedCollection references . . . 229
58. IBMTSSVC_AuthorizedStorageHardwareID references . . . 229
59. IBMTSSVC_AuthorizedSubject references . . . 230
60. IBMTSSVC_AuthorizationTarget references . . . 230
61. IBMTSSVC_AvailableHardwareID references . . . 231
62. IBMTSSVC_BackendSCSILUN references . . . 231
63. IBMTSSVC_BasedOn references . . . 231
64. IBMTSSVC_BasedOn properties . . . 231
65. IBMTSSVC_ClusterController references . . . 232
66. IBMTSSVC_ClusterDumps references . . . 232
67. IBMTSSVC_ClusterDumps properties . . . 233
68. IBMTSSVC_ClusteringCandidate references . . . 233
69. IBMTSSVC_ClusteringServiceForSystem references . . . 233
70. IBMTSSVC_ClusterMaskingCapabilities references . . . 234
71. IBMTSSVC_ClusterPort references . . . 234
72. IBMTSSVC_ClusterScopeCandidateVolume references . . . 235
73. IBMTSSVC_ClusterScopeChassis references . . . 235
74. IBMTSSVC_ClusterScopeFCSet references . . . 235
75. IBMTSSVC_ClusterScopeIOGroup references . . . 235
76. IBMTSSVC_ClusterScopeNodeVPD references . . . 236
77. IBMTSSVC_ClusterScopePrivilege references . . . 236
78. IBMTSSVC_ClusterScopeProduct references . . . 236
79. IBMTSSVC_ClusterScopeSCSet references . . . 237
80. IBMTSSVC_ClusterVolume references . . . 237
81. IBMTSSVC_ComponentCS references . . . 237
82. IBMTSSVC_ComputerSystemPackage references . . . 238
83. IBMTSSVC_ComputerSystemPackage properties . . . 238
84. IBMTSSVC_ConnectedBackendController references . . . 238
85. IBMTSSVC_ControllerConfigurationServiceForSystem references . . . 238
86. IBMTSSVC_ControllerConfService MaskingCapabilities references . . . 239
87. IBMTSSVC_CopyCandidate references . . . 239
88. IBMTSSVC_ComponentCS references . . . 240
89. IBMTSSVC_FlashCopyStorageSynchronized references . . . 240
90. IBMTSSVC_FlashCopyStorageSynchronized properties . . . 240
91. IBMTSSVC_FlashCopySynchronizedMember references . . . 242
92. IBMTSSVC_HardwareIDOnSystem references . . . 243
93. IBMTSSVC_HostedAccessPoint references . . . 243
94. IBMTSSVC_HostedJob references . . . 243
95. IBMTSSVC_HostedPrimordialPool references . . . 244
96. IBMTSSVC_HostedStoragePool references . . . 244
97. IBMTSSVC_HwIDCollectionOnSystem references . . . 244
98. IBMTSSVC_IndicationFiltersConformsToProfile references . . . 245
99. IBMTSSVC_IndicationFiltersConformsToSubProfile references . . . 245
100. IBMTSSVC_IOGroupIdentity references . . . 245
101. IBMTSSVC_IOGroupPort references . . . 246
102. IBMTSSVC_ManagesCollection references . . . 246
103. IBMTSSVC_ManagesController references . . . 246
104. IBMTSSVC_ManagesHardwareID references . . . 247
105. IBMTSSVC_ManagesPrivilege references . . . 247
106. IBMTSSVC_MemberOfCollection references . . . 247
107. IBMTSSVC_MemberOfIOGroup references . . . 248
108. IBMTSSVC_NodeDumps references . . . 248
109. IBMTSSVC_NodeDumps . . . 248

110. IBMTSSVC_PartnershipCandidate references . . . 249
111. IBMTSSVC_PoolCapabilities references . . . 249
112. IBMTSSVC_PrimordialPoolCapabilities references . . . 249
113. IBMTSSVC_PrimordialPoolComponent references . . . 250
114. IBMTSSVC_PrimordialPoolForController references . . . 250
115. IBMTSSVC_PrivilegeServiceForSystem references . . . 250
116. IBMTSSVC_ProductPhysicalComponent references . . . 251
117. IBMTSSVC_ProtocolControllerForPort references . . . 251
118. IBMTSSVC_ProtocolControllerForPort . . . 251
119. IBMTSSVC_ProtocolControllerForUnit references . . . 252
120. IBMTSSVC_ProtocolControllerForUnit . . . 253
121. IBMTSSVC_ProviderInObjectManager references . . . 253
122. IBMTSSVC_RemotePartnership references . . . 254
123. IBMTSSVC_RemoteSystemVolume references . . . 254
124. IBMTSSVC_RequiresProfile references . . . 254
125. IBMTSSVC_SAPAvailableForElement references . . . 255
126. IBMTSSVC_StorageConfigurationServiceCapabilities references . . . 255
127. IBMTSSVC_StorageConfigurationServiceForSystem references . . . 255
128. IBMTSSVC_StorageHardwareIDManagementServiceForSystem references . . . 256
129. IBMTSSVC_StoragePoolComponent references . . . 256
130. IBMTSSVC_SyncCopyStorageSynchronized references . . . 256
131. IBMTSSVC_SyncCopyStorageSynchronized properties . . . 257
132. IBMTSSVC_SyncCopySynchronizedMember references . . . 260
133. IBMTSSVC_SystemBackendVolume references . . . 260
134. IBMTSSVC_SystemCandidateVolume references . . . 260
135. IBMTSSVC_SystemController references . . . 261
136. IBMTSSVC_SystemFCPort references . . . 261
137. IBMTSSVC_SystemFeatures references . . . 261
138. IBMTSSVC_SystemVolume references . . . 262
139. IBMTSSVC_SystemVPD references . . . 262
140. IBMTSSVC_SystemVPD properties . . . 262
141. IBMTSSVC_UseOfMessageLog references . . . 263
142. IBMTSSVC_UseOfMessageLog properties . . . 263
143. IBMTSSVC_VolumeSettingData references . . . 263
144. IBMTSSVC_VolumeSettingData properties . . . 264
145. IBMTSSVC_AccountManagementServiceForSystem references . . . 264
146. IBMTS_AccountOnCIMOM references . . . 265
147. IBMTS_AccountOnCIMOM properties . . . 265
148. IBMTS_AccountOnSystem . . . 265
149. IBMTS_AccountOnSystem properties . . . 266
150. IBMTS_CommMechanismForManager . . . 266
151. IBMTS_ElementConformsToProfile . . . 266
152. IBMTS_HostedAccessPoint . . . 267
153. IBMTS_HostedService . . . 267
154. IBMTS_IndicationFiltersConformsToProfile . . . 267
155. IBMTS_ManagesAccount references . . . 268
156. IBMTS_NamespaceInManager . . . 268
157. Supported intrinsic methods . . . 269
158. Associators() parameters . . . 270
159. AssociatorNames() parameters . . . 271
160. CreateInstance() parameters . . . 271
161. DeleteInstance() parameters . . . 272
162. EnumerateClasses() parameters . . . 272
163. EnumerateClassNames() parameters . . . 273
164. EnumerateInstances() parameters . . . 274
165. EnumerateInstanceNames() parameters . . . 274

166. ExecQuery() parameters 275
167. GetClass() parameters 275
168. GetInstance() parameters 276
169. GetProperty() parameters 277
170. ModifyInstance() parameters 277
171. References() parameters 278
172. ReferenceNames() parameters 278
173. SetProperty() parameters 279
174. Supported extrinsic methods 280
175. Add2062Cluster() parameters. 282
176. Add2145Cluster() parameters. 283
177. AddHardwareIDsToCollection() parameters 283
178. AddNode() parameters 284
179. AssignAccess() parameters 285
180. AttachDevice() parameters 287
181. AttachReplica() parameters 287
182. BackupConfiguration() parameters 289
183. CancelItration() parameters 289
184. Clean() parameters 290
185. Create2062Cluster() parameters. 291
186. CreateHardwareIDCollection() parameters 292
187. CreateOrModifyStoragePool() parameters 293
188. CreateOrModifyElementFromStoragePool() parameters 295
189. CreateProtocolControllerWithPorts() parameters 297
190. CreateRemoteClusterPartnership() parameters 298
191. CreateReplica() parameters 299
192. CreateSetting() parameters 300
193. CreateStorageHardwareID() parameters 300
194. CreateSynchronizedSet() parameters 301
195. DeleteConfigurationBackup() parameters 302
196. DeleteHardwareIDCollection() parameters 303
197. DeleteProtocolController() parameters 303
198. DeleteRecord() parameters 304
199. DeleteRemoteClusterPartnership() parameters 305
200. DeleteStorageHardwareID() parameters 305
201. DeleteSynchronizedSet() 306
202. DeleteStoragePool() 307
203. DetachDevice() parameters 308
204. Dump() parameters 308
205. Enter() parameters. 309
206. EvictNode() parameters 310
207. Exit() parameters 310
208. FixRecord() parameters 311
209. GetAllRecords() parameters 311
210. GetDump() parameters 312
211. GetFreeExtents() parameters 312
212. GetRecord() parameters. 313
213. GetResetPasswordChangeFeatureStatus() properties 314
214. GetSupportedSizeRange() parameters 314
215. GetSupported Sizes() parameters 315
216. IncludeBackendVolume() parameters 315
217. ListConfigurationBackups() parameters 316
218. MigrateVolume() parameters 316
219. ModifyErrorSettings() parameters 317
220. Modifyipaddress() parameters 318
221. ModifyResetPasswordChangeFeature() parameters 318

222. ModifySynchronization() parameters 319
223. ModifySynchronizedSet() parameters 321
224. PositionAtRecord() parameters 323
225. PositionToFirstRecord() parameters 323
226. PositionToFirstRecordRoot() parameters 324
227. PositionToFirstRecordType() parameters 324
228. Reload2062Node() parameters 325
229. RemoveAccess() parameters 326
230. RemoveCluster() parameters 327
231. RequestDiscovery() parameters 327
232. Reset2062Node() parameters. 328
233. RestoreConfiguration() parameters 328
234. ReturnToStoragePool() parameters. 329
235. SetIOGroup() parameters 330
236. SetLocale() parameters 330
237. SetPasswords() parameters 331
238. SetQuorum() parameters 331
239. SetTimeZone() parameters. 332
240. StartStatisticsCollection() parameters 332
241. Shutdown() parameters 333
242. UnfixRecord() parameters 334
243. Upgrade() parameters 334
244. WriteRecord() parameters 335

About this guide

This publication introduces the Common Information Model (CIM) Agent for the IBM® TotalStorage™ SAN Volume Controller.

This section describes:

- Content and intended audience of this book
- Typefaces that are used to show emphasis
- Information that is related to this book
- How to order IBM publications
- How to send in your feedback on this book
- Web sites that provide information about the SAN Volume Controller or related products or technologies

Who should use this guide?

This reference book is for application programmers who are developing with the Common Information Model (CIM).

This reference book is for CIM-based application programmers who want to do the following tasks:

- Understand the CIM Agent for the SAN Volume Controller
- Discover and connect to the CIM Agent service
- Retrieve and extract the CIM Agent object classes, attributes, and methods
- Create new object instances for basic storage configuration, LUN masking, and copy services on the SAN Volume Controller.

Summary of changes

This document contains terminology, maintenance, and editorial changes.

Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change. This summary of changes describes new functions that have been added to this release.

Summary of changes for GC26-7545-02 SAN Volume Controller CIM Agent Developer's Reference Release 1.2.1

The Summary of Changes provides a list of new, modified, and changed information since the last version of the reference

New information

This topic describes the changes to this reference since the previous edition, SC26-7545-01.

- Full list of attributes provided through the Service Location Protocol (SLP)
- Manual CIMOM IP registration
- Manual setting of the RemoteServiceAccessPoint connection data

© Copyright IBM Corp. 2003, 2004

Changed information

No information has been changed in this version.

Deleted information

No deletions were made in this version of the reference.

Emphasis

Different typefaces are used in this guide to show emphasis.

The following typefaces are used to show emphasis:

Table 1. Emphasis descriptions

Boldface	Text in **boldface** represents menu items and command names.
Italics	Text in *italics* is used to emphasize a word. In command syntax, it is used for variables for which you supply actual values, such as a default directory or the name of a cluster.
`Monospace`	Text in `monospace` identifies the data or commands that you type, samples of command output, examples of program code or messages from the system, or names of command flags, parameters, arguments, and name-value pairs.

SAN Volume Controller library and related publications

A list of other publications that are related to this product are provided to you for your reference.

The tables in this section list and describe the following publications:

- The publications that make up the library for the IBM TotalStorage SAN Volume Controller
- Other IBM publications that relate to the SAN Volume Controller

SAN Volume Controller library

Table 2 on page xix lists and describes the publications that make up the SAN Volume Controller library. Unless otherwise noted, these publications are available in Adobe portable document format (PDF) on a compact disc (CD) that comes with the SAN Volume Controller. If you need additional copies of this CD, the order number is SK2T-8811. These publications are also available as PDF files from the following Web site:

http://www-1.ibm.com/servers/storage/support/virtual/2145.html

Table 2. Publications in the SAN Volume Controller library

Title	Description	Order number
IBM TotalStorage SAN Volume Controller: CIM Agent Developer's Reference	This reference guide describes the objects and classes in a Common Information Model (CIM) environment.	SC26-7590
IBM TotalStorage SAN Volume Controller: Command-Line Interface User's Guide	This guide describes the commands that you can use from the SAN Volume Controller command-line interface (CLI).	SC26-7544
IBM TotalStorage SAN Volume Controller: Configuration Guide	This guide provides guidelines for configuring your SAN Volume Controller.	SC26-7543
IBM TotalStorage SAN Volume Controller: Host Attachment Guide	This guide provides guidelines for attaching the SAN Volume Controller to your host system.	SC26-7575
IBM TotalStorage SAN Volume Controller: Installation Guide	This guide includes the instructions the service representative uses to install the SAN Volume Controller.	SC26-7541
IBM TotalStorage SAN Volume Controller: Planning Guide	This guide introduces the SAN Volume Controller and lists the features you can order. It also provides guidelines for planning the installation and configuration of the SAN Volume Controller.	GA22-1052
IBM TotalStorage SAN Volume Controller: Service Guide	This guide includes the instructions the service representative uses to service the SAN Volume Controller.	SC26-7542
IBM TotalStorage SAN Volume Controller: Translated Safety Notices	This guide contains the danger and caution notices for the SAN Volume Controller. The notices are shown in English and in numerous other languages.	SC26-7577

Other IBM publications

Table 3 on page xx lists and describes other IBM publications that contain additional information related to the SAN Volume Controller.

Table 3. Other IBM publications

Title	Description	Order number
IBM TotalStorage Enterprise Storage Server, IBM TotalStorage SAN Volume Controller, IBM TotalStorage SAN Volume Controller for Cisco MDS 9000, Subsystem Device Driver: User's Guide	This guide describes the IBM Subsystem Device Driver Version 1.5 for TotalStorage Products and how to use it with the SAN Volume Controller. This publication is referred to as the *IBM TotalStorage Subsystem Device Driver: User's Guide*.	SC26-7608

Related Web sites

Table 4 lists Web sites that have information about SAN Volume Controller or related products or technologies.

Table 4. Web sites

Type of information	Web site
SAN Volume Controller support	http://www-1.ibm.com/servers/storage/support/virtual/2145.html
Technical support for IBM storage products	http://www.ibm.com/storage/support/

How to order IBM publications

The publications center is a worldwide central repository for IBM product publications and marketing material.

The IBM publications center

The IBM publications center offers customized search functions to help you find the publications that you need. Some publications are available for you to view or download free of charge. You can also order publications. The publications center displays prices in your local currency. You can access the IBM publications center through the following Web site:

www.ibm.com/shop/publications/order/

Publications notification system

The IBM publications center Web site offers you a notification system for IBM publications. Register and you can create your own profile of publications that interest you. The publications notification system sends you a daily e-mail that contains information about new or revised publications that are based on your profile.

If you want to subscribe, you can access the publications notification system from the IBM publications center at the following Web site:

www.ibm.com/shop/publications/order/

How to send your comments

Your feedback is important to help us provide the highest quality information. If you have any comments about this book or any other documentation, you can submit them in one of the following ways:

- e-mail

  Submit your comments electronically to the following e-mail address:

  starpubs@us.ibm.com

  Be sure to include the name and order number of the book and, if applicable, the specific location of the text you are commenting on, such as a page number or table number.

- Mail

  Fill out the Readers' Comments form (RCF) at the back of this book. If the RCF has been removed, you can address your comments to:

  International Business Machines Corporation
  RCF Processing Department
  Department 61C
  9032 South Rita Road
  Tucson, Arizona 85775-4401
  U.S.A.

Chapter 1. Introduction

Introducing the Common Information Model.

This chapter introduces the Common Information Model (CIM) Agent for SAN Volume Controller. It provides overviews of the:

- Storage Management Initiative Specification (SMI-S)
- CIM
- CIM-related concepts
- CIM agent
- SAN Volume Controller
- CIM Agent for the SAN Volume Controller

This chapter also presents functional views of the CIM Agent object models.

Storage Management Initiative Specification

Description of the Storage Management Initiative Specification.

The Storage Management Initiative Specification (SMI-S) is a design specification of the Storage Management Initiative (SMI) launched by the Storage Networking Industry Association (SNIA). It specifies a secure and reliable interface that allows storage management systems to identify, classify, monitor, and control physical and logical resources in a storage area network (SAN). The interface is intended as a solution that integrates the various devices to be managed in a SAN and the tools used to manage them.

SMI-S is based on a number of existing technologies or industry standards that include the following:

Common Information Model (CIM)
An object model for data storage and management developed by the Distributed Management Task Force (DMTF). CIM makes it possible to organize devices and components of devices in an object-oriented pattern.

Web-Based Enterprise Management (WBEM)
A tiered enterprise management architecture also developed by the DMTF. This architecture provides the management design framework that consists of devices, device providers, the object manager, and the messaging protocol for the communication between client applications and the object manager. In the case of the CIM, the object manager is the CIMOM and the messaging protocol is the CIM over HTTP technology. The CIM over HTTP approach specifies that the CIM data is encoded in XML and sent in specific messages between the client applications and the CIMOM over the TCP/IP network in a SAN.

Service Location Protocol (SLP)
A directory service that the client application calls to locate the CIMOM.

Intended to be an industry standard, SMI-S extends the generic capabilities of the CIM, the WBEM, and the SLP to implement storage networking interoperability. For example, the WBEM is expanded to provide provisions for security, resource-locking management, event notification, and service discovery.

© Copyright IBM Corp. 2003, 2004

Common Information Model

Description of the Common Information Model.

The Common Information Model (CIM) is a set of standards developed by the Distributed Management Task Force (DMTF). It provides an open approach to the design and implementation of storage systems, applications, databases, networks, and devices.

The CIM specifications provide the language and the methodology for describing management data. For example, CIM Schema 2.7 for Managing Storage Arrays specifies how the management environment should be enabled for data management in a common way. Specifically, the CIM defines common object classes, associations, and methods. Member vendors can use those objects and extend them to specify how data should be processed and organized in a specific managed environment.

Common Information Model concepts

Concepts of the Common Information Model.

The Common Information Model (CIM) specifications use the following concepts and terminology to describe the object models:

Association
A class that contains two references that define a relationship between two referenced objects.

Class The definition of an object within a specific hierarchy. An object class can have properties and methods and serve as the target of an association.

Indication
An object representation of an event.

Instance
An individual object that is the member of some class. In object-oriented programming, an object created by instantiating a class.

Method
A way to implement a function on a class.

Namespace
The scope within which a CIM schema applies.

Object path
An object that consists of a namespace path and a model path. The namespace path provides access to the CIM implementation managed by the CIM Agent, and the model path provides navigation within the implementation.

Property
An attribute that is used to characterize instances of a class.

Qualifier
A value that provides additional information about a class, association, indication, method, method parameter, instance, property, or reference.

Reference
A pointer to another instance that defines the role and scope of an object in an association.

Schema
A group of object classes defined for and applicable to a single namespace. Within the CIM Agent, the supported schemas are the ones that are loaded through the managed object format (MOF) compiler.

CIM agent

With a Common Information Model (CIM) agent, programmers can use common building blocks rather than proprietary software or device-specific programming interfaces to manage CIM-compliant devices. Standardization of the way applications handle storage provides easier storage management.

Components

A CIM agent typically involves the following components:

agent code
An open-systems standard that interprets CIM requests and responses as they transfer between the client application and the device.

CIM object manager (CIMOM)
The common conceptual framework for data management that receives, validates, and authenticates the CIM requests from the client application. It then directs the requests to the appropriate component or device provider.

client application
A storage management program that initiates CIM requests to the CIM agent for the device.

device
The storage server that processes and hosts the client application requests.

device provider
A device-specific handler that serves as a plug-in for the CIM. That is, the CIMOM uses the handler to interface with the device.

Service Location Protocol (SLP)
A directory service that the client application calls to locate the CIMOM.

CIM agent at work

Figure 1 on page 4 shows the way a typical CIM agent works. The client application locates the CIMOM by calling an SLP directory service. When the CIMOM is first invoked, it registers itself to the SLP Service Agent and supplies its location, IP address, port number, and the type of service it provides. A string describing the CIM Agents access point is registered. The following shows an example of the registered string:

```
service:wbem:https://<CIM Agent IP>:<port number>
```

Note: There are two standard ports for SMI-S: 5989 and 5988. Port 5989 is used for secure communication and port 5988 is used for unsecure communication.

The SLP provides the following attributes:

```
service:wbem:https://<CIM Agent IP>:5989

service-hi-name=SVC CIM Agent 1.2.1.0

service-hi-description=IBM SAN Volume Controller CIM Agent Version 1.2.1
```

```
service-location-tcp=https://<CIM Agent IP>:5989

service-id=IBM_CIMOM_1094736587984_1108027145

template-url-syntax=https://<CIM Agent IP>

CommunicationMechanism=cim-xml

InteropSchemaNamespace=/root/ibm

FunctionalProfilesSupported=Basic Read, Basic Write, Instance Manipulation,
Association Traversal, Query Execution, Qualifier Declaration, Indications

MultipleOperationsSupported=false

ProtocolVersion=1

AuthenticationMechanismSupported=Basic

Namespace=/root/ibm
```

With this information, the client application starts to directly communicate with the CIMOM.

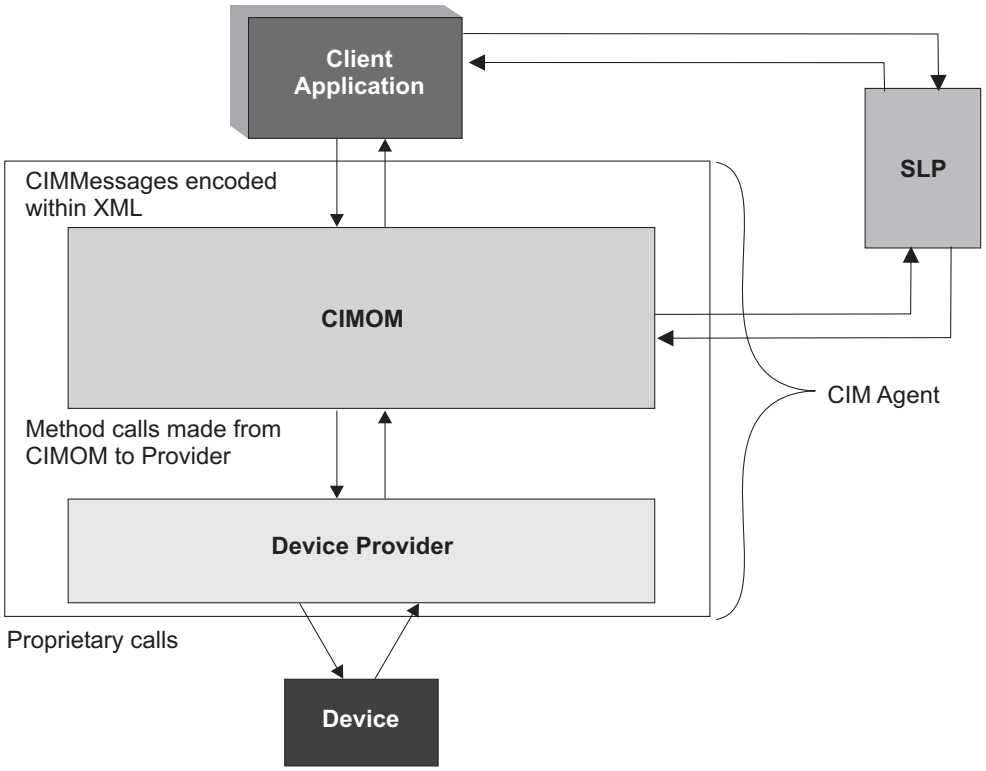

Figure 1. A typical CIM Agent at work

The client application then sends CIM requests to the CIMOM. As requests arrive, the CIMOM validates and authenticates each request. It then directs the requests to the appropriate functional component of the CIMOM or to a device provider. The provider makes calls to a device-unique programming interface on behalf of the CIMOM to satisfy client application requests.

The management application can obtain an instance of the RemoteServiceAccessPoint from the CIMOM. This instance allows the management application to access the Web User Interface.

SAN Volume Controller

The SAN Volume Controller is a SAN appliance that attaches open-systems storage devices to supported open-systems hosts.

The IBM TotalStorage SAN Volume Controller provides symmetric virtualization by creating a pool of managed disks from the attached storage subsystems, which are then mapped to a set of virtual disks for use by attached host computer systems. System administrators can view and access a common pool of storage on the SAN, which enables them to use storage resources more efficiently and provides a common base for advanced functions.

The SAN Volume Controller is analogous to a logical volume manager (LVM) on a SAN. It performs the following functions for the SAN storage that it is controlling:

- Creates a single pool of storage
- Manages logical volumes
- Provides advanced functions for the SAN, such as:
  - Large scalable cache
  - Copy services
    - Point-in-time Copy
    - FlashCopy® (point-in-time copy)
    - Remote Copy (synchronous copy)
    - Data migration
  - Space management
    - Mapping that is based on desired performance characteristics
    - Quality of service metering

A *node* is a single storage engine. See Figure 2 on page 6 for a visual of a node. The storage engines are always installed in pairs with one to four pairs of nodes constituting a *cluster*. Each node in a pair is configured to back up the other. Each pair of nodes is known as an *I/O group*. All I/O operations that are managed by the nodes in an I/O group are cached on both nodes for resilience. Each virtual volume is defined to an I/O group. To avoid any single point of failure, the nodes of an I/O group are protected by independent uninterruptible power supply units.

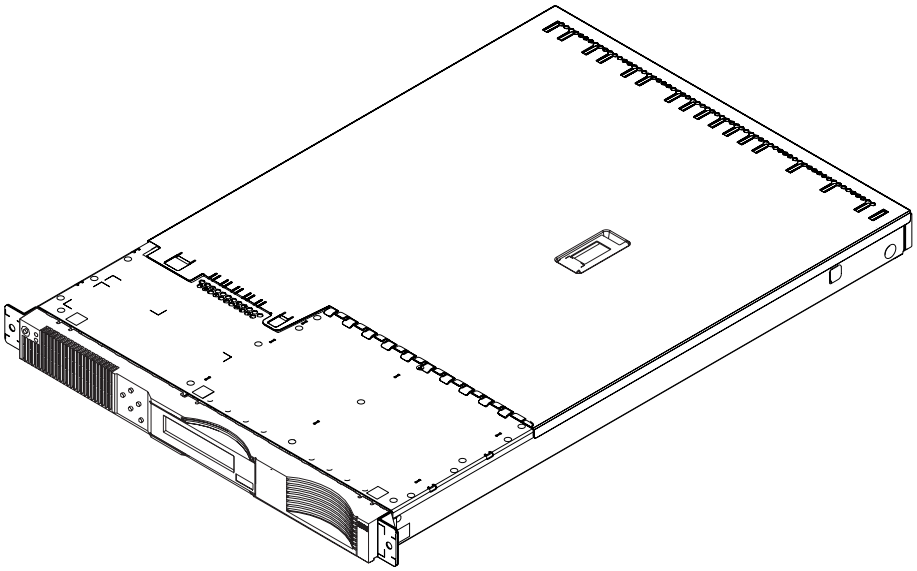

Figure 2. A SAN Volume Controller node

The SAN Volume Controller I/O groups see the storage presented to the SAN by the backend controllers as a number of disks known as *managed disks*. The application services do not see these managed disks. Instead they see a number of logical disks, known as *virtual disks*, that are presented to the SAN by the SAN Volume Controller. Each node must only be in one I/O group and provide access to the virtual disks in the I/O group.

The SAN Volume Controller helps to provide continuous operations and can also optimize the data path to ensure performance levels are maintained. Ensure that you use IBM TotalStorage Multiple Device Manager performance manager to analyze the performance statistics. See *IBM TotalStorage Multiple Device Manager Configuration and Installation Guide* and *IBM TotalStorage Multiple Device Manager CLI Guide* for more information.

The fabric contains two distinct zones: a host zone and a disk zone. In the host zone, the host systems can identify and address the nodes. You can have more than one host zone. Generally, you will create one host zone per operating system type. In the disk zone, the nodes can identify the disk drives. Host systems cannot operate on the disk drives directly; all data transfer occurs through the nodes. As shown in Figure 3 on page 7, several host systems can be connected to a SAN fabric. A cluster of SAN Volume Controllers is connected to the same fabric and presents virtual disks to the host systems. You configure these virtual disks using the disks located on the RAID controllers.

Note: You can have more than one host zone. Generally you create one host zone per operating system type because some operating systems will not tolerate other operating systems in the same zone.

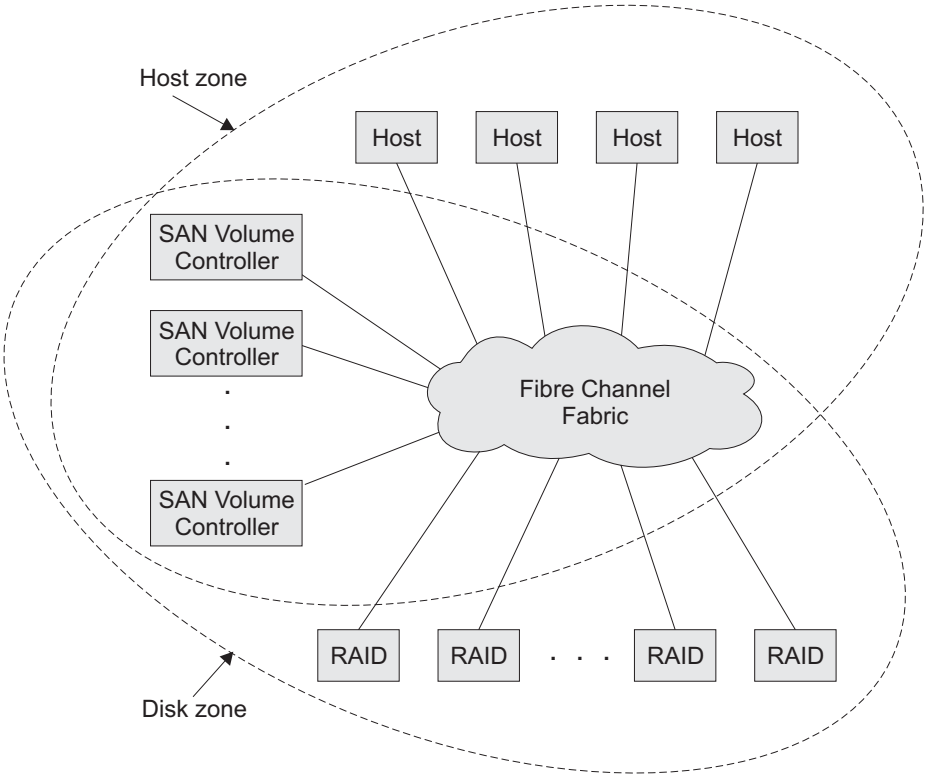

Figure 3. Example of a SAN Volume Controller in a fabric

You can remove one node in each I/O group from a cluster when hardware service or maintenance is required. After you remove the node, you can replace the field replaceable units (FRUs) in the node. All disk drive communication and communication between nodes is performed through the SAN. All SAN Volume Controller configuration and service commands are sent to the cluster through an Ethernet network.

Each node contains its own vital product data (VPD). Each cluster contains VPD that is common to all the nodes on the cluster, and any system connected to the Ethernet network can access this VPD.

Cluster configuration information is stored on every node that is in the cluster to allow concurrent replacement of FRUs. An example of this information might be information that is displayed on the menu screen of the SAN Volume Controller. When a new FRU is installed and when the node is added back into the cluster, configuration information that is required by that node is read from other nodes in the cluster.

SAN Volume Controller operating environment

- Minimum of one pair of SAN Volume Controller nodes
- Minimum two uninterruptible power supplies
- One master console is required per SAN installation for configuration

Features of a SAN Volume Controller node

- 19-inch rack mounted enclosure
- 4 fibre channel ports
- 2 fibre channel adapters

- 4 GB cache memory

Supported hosts

For a list of supported operating systems, see the IBM TotalStorage SAN Volume Controller Web site at:

http://www-1.ibm.com/servers/storage/support/virtual/2145.html

Multipathing software

- IBM Subsystem Device Driver (SDD)
- Redundant Dual Active Controller (RDAC)

Note: Direct attach hosts sharing a back end storage controller with a SAN Volume Controller can run multipath drivers SDD and RDAC. There is no support for the co-existence of native multipath drivers with SDD on the same host.

Check the following Web site for the latest support and coexistence information:

http://www-1.ibm.com/servers/storage/support/virtual/2145.html

User interfaces

The SAN Volume Controller provides the following user interfaces:

- IBM TotalStorage SAN Volume Controller Console, a Web-accessible graphical user interface (GUI) that supports flexible and rapid access to storage management information
- A command-line interface (CLI) using Secure Shell (SSH)

Application programming interfaces

The SAN Volume Controller provides the following application programming interface:

- IBM TotalStorage Common Information Model (CIM) Agent for the SAN Volume Controller, which supports the Storage Management Initiative Specification of the Storage Network Industry Association.

Common Information Model Agent for the SAN Volume Controller

The Common Information Model (CIM) Agent for the SAN Volume Controller serves as a configuration interface to the SAN Volume Controller.

It consists of the following main components:

- CIM object manager (CIMOM)
- Service Location Protocol (SLP) Agent
- SAN Volume Controller provider

The SAN Volume Controller Console is configured to locate the CIMOM though its IP address. When the CIMOM is started, it registers itself with the SLP directory service by supplying its IP address, port number, and service type information. With the location information secured, the SAN Volume Controller Console begins to communicate directly with the CIMOM and the SAN Volume Controller provider. The provider knows how to use functionality provided by the SAN Volume Controller upon the requests from the CIMOM.

Validating the truststore certificate expiration

In order to successfully log onto the master console, you must ensure that you have a valid truststore certificate.

When signing onto the master console, you might receive a message similar to the following:

CMMUI8304E The Administrative server is unable to find a valid certificate in the truststore file.

This message is displayed when a certificate in the truststore file expires. The Administrative server uses the certificates in the truststore file to create a secure connection with the CIM agent. Because the Administrative server cannot find a valid certificate for the CIM agent in the truststore file, no authentication can occur.

To resolve the problem you must verify that the truststore file was created correctly. If you have any problems, contact your service representative.

Perform the following steps to regenerate a truststore certificate:

1. Go to the C:\Program Files\IBM\svcconsole\cimom directory.
2. Double-click on the **mkcertificate.bat** file. A ″Generating Certificates″ message is displayed. The new certificate is generated and stored in the C:\Program Files\IBM\svcconsole\cimom directory.
3. Copy the truststore file to the following sub directories:

   Note: Each directory begins with C:\Program Files\IBM\svcconsole\console\embeddedWAS...

   C:\...\config\cells\DefaultNode\applications\
   ICAConsole.ear\deployments\ICAConsole\ICAConsole.war\
   WEB-INF

   C:\...\config\cells\DefaultNode\applications\
   SVCConsole.ear\deployments\SVCConsole\SVCConsole.war\
   WEB-INF

   C:\...\config\installedApps\DefaultNode\
   ICAConsole.ear\ICAConsole.war\WEB-INF

   C:\...\config\installedApps\DefaultNode\
   SVCConsole.ear\SVCConsole.war\WEB-INF
4. Stop and then restart the following applications. The following services are located in **Start -> Settings -> Control Panel -> Administrative Tools -> Component Services**.
   - IBM CIM Object Manager
   - IBM WebSphere Application Server V5 - SVC

   To stop and then restart the services, right-click on the application and select **Stop**, then **Start**.

   Note: If the command times-out when stopping the IBM WebSphere application, you can restart the master console as this will restart the application as well.

5. Ensure that both applications are running again. Launch theSAN Volume Controller Console and logon.

Functional views of the Common Information Model Agent

Functional views of the Common Information Model (CIM) Agent object model.

The following topics provide functional views of the Common Information Model (CIM) Agent object model. These diagrams show specific functionality that the CIM Agent provides, including storage configuration service, Copy Services, LUN masking, and security, and illustrate the architecture of the CIM Agent for the SAN Volume Controller.

Profile Overview

The graphic below provides a high-level overview of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 4 on page 11 shows the SMI-S profiles and subprofiles supported.

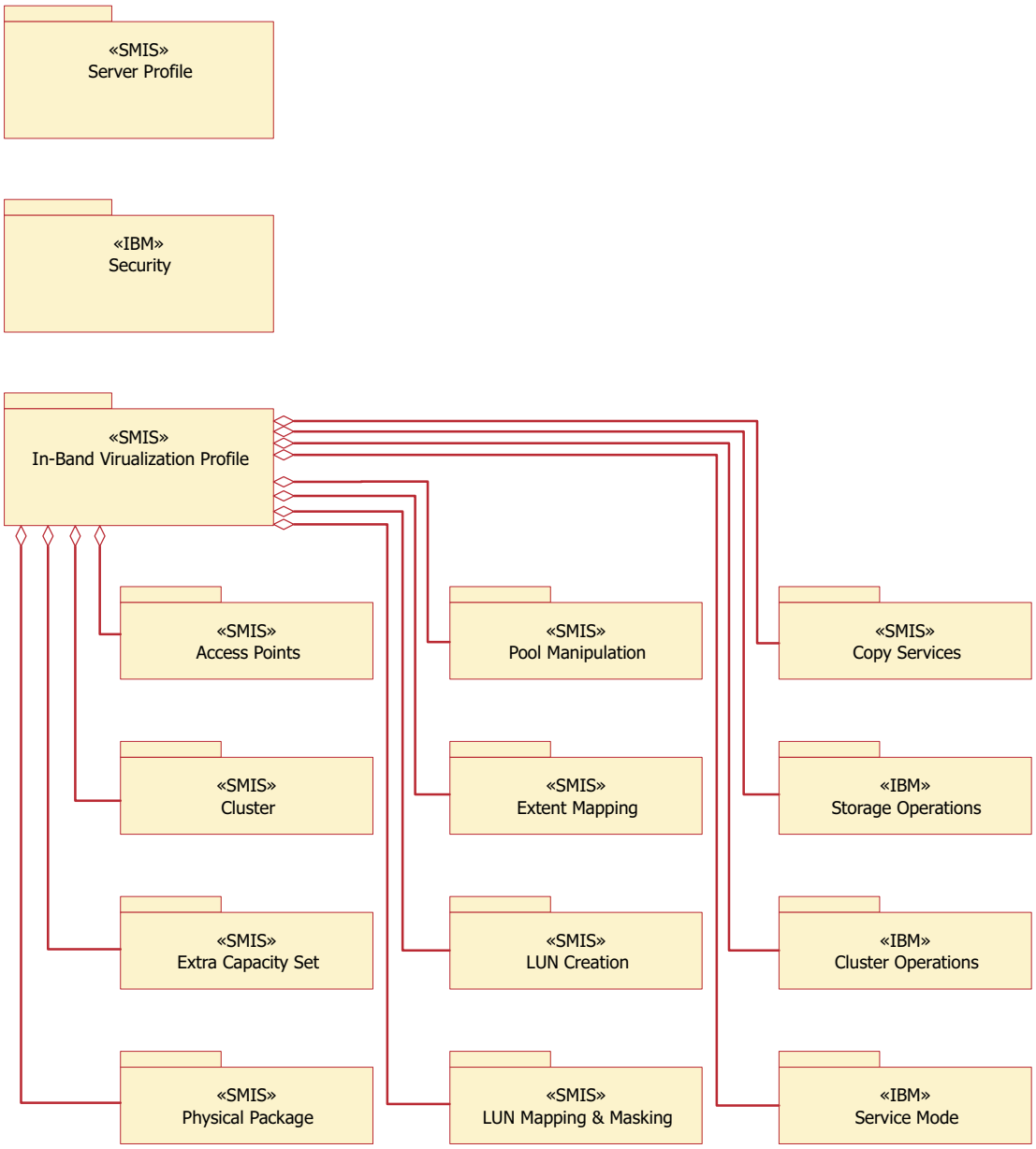

Figure 4. High-level overview of the CIM Agent for the SAN Volume Controller.

Physical Package

The graphic below provides a high-level overview of the physical package of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 5 on page 12 shows the basic classes (building blocks) for the model.

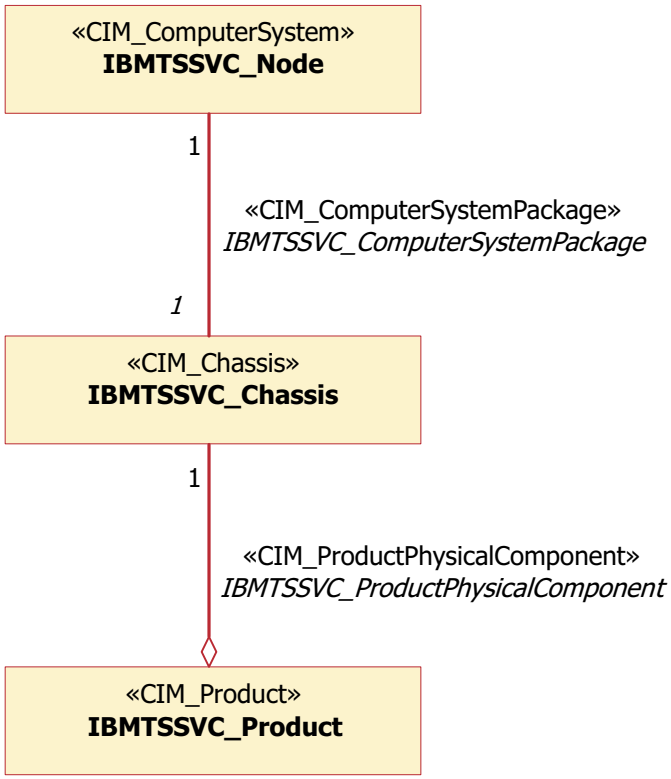

Figure 5. High-level overview of the physical package of the CIM Agent for the SAN Volume Controller.

Server Profile

The graphic below provides a high-level overview of the server profile of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 6 on page 13 shows the basic classes (building blocks) for the model.

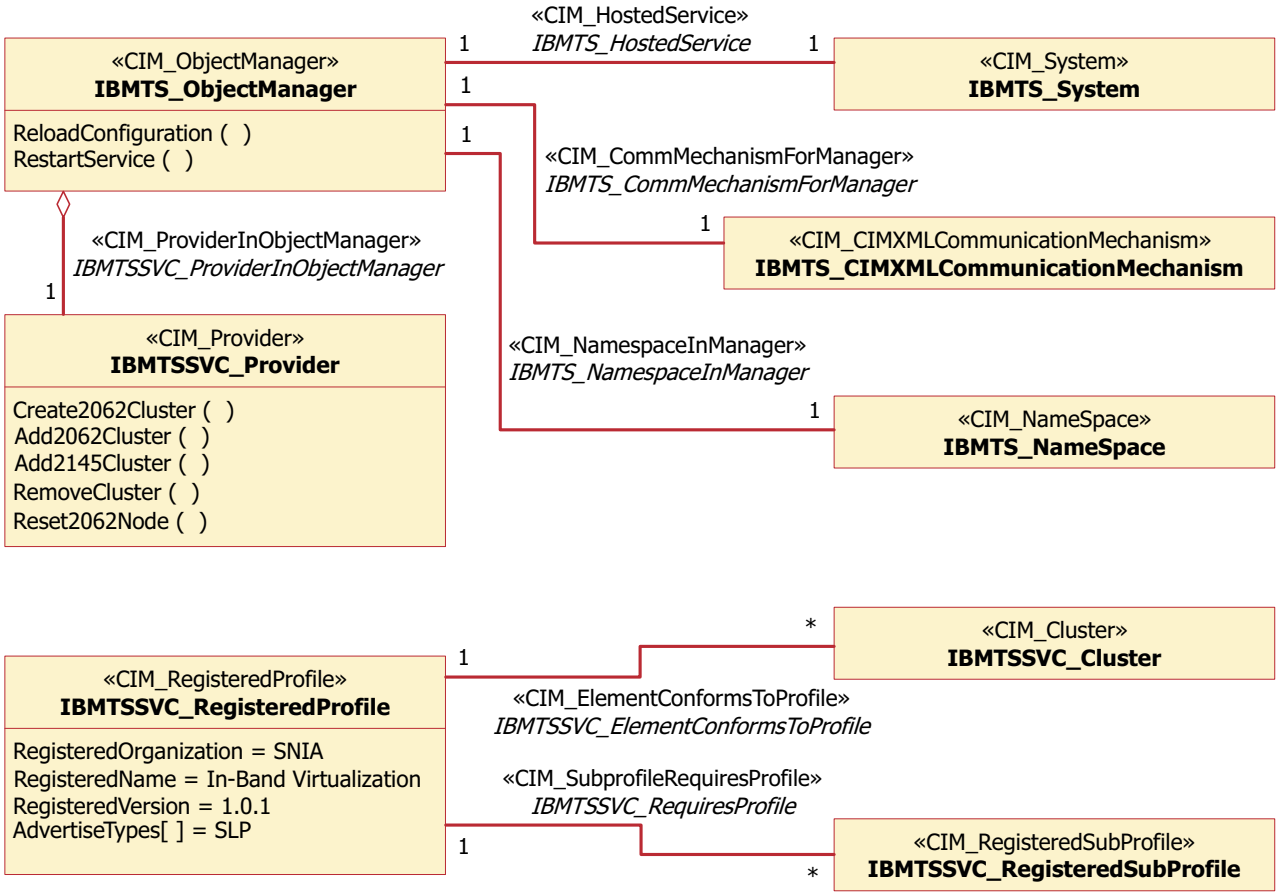

Figure 6. High-level overview of the server profile of the CIM Agent for the SAN Volume Controller.

Extent mapping subprofile

The graphic below provides a high-level overview of the extent mapping subprofile of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 7 on page 14 shows the basic classes (building blocks) for the model.

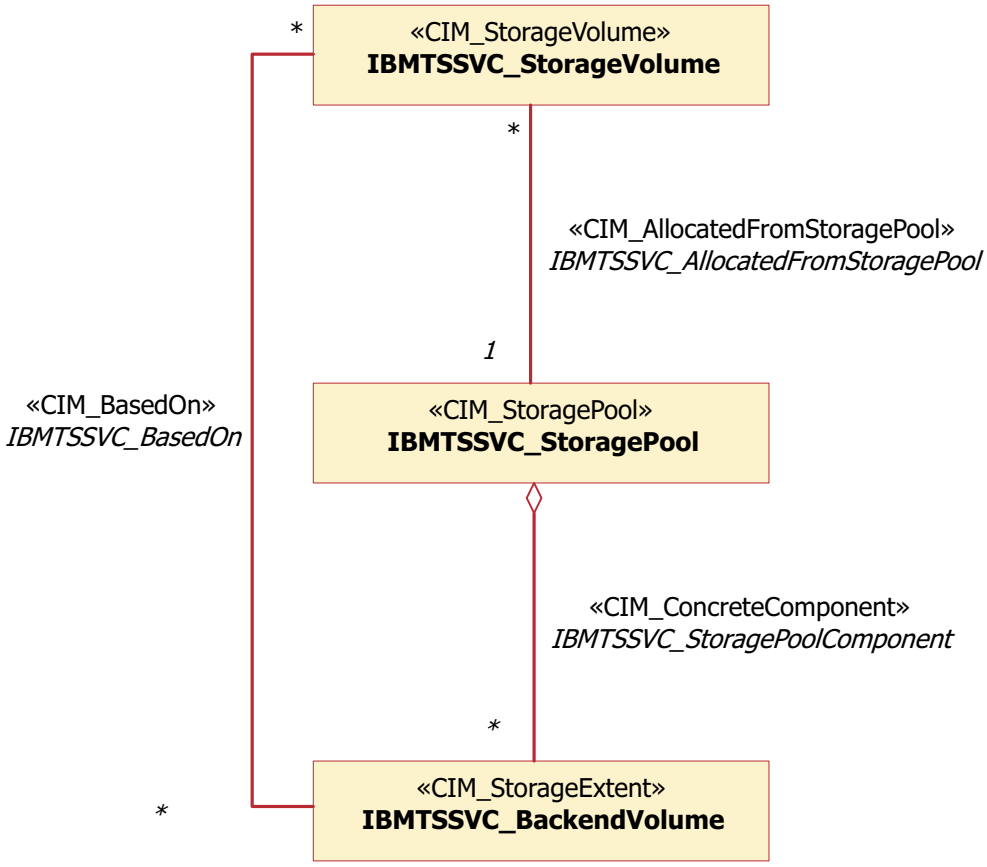

Figure 7. High-level overview of the extent mapping subprofile of the CIM Agent for the SAN Volume Controller.

ExtraCapacitySet subprofile

The graphic below provides a high-level overview of the ExtraCapacitySet subprofile of the Common Information Model Agent for the SAN Volume Controller.

Figure 8 on page 15 shows the basic classes (building blocks) for the model.

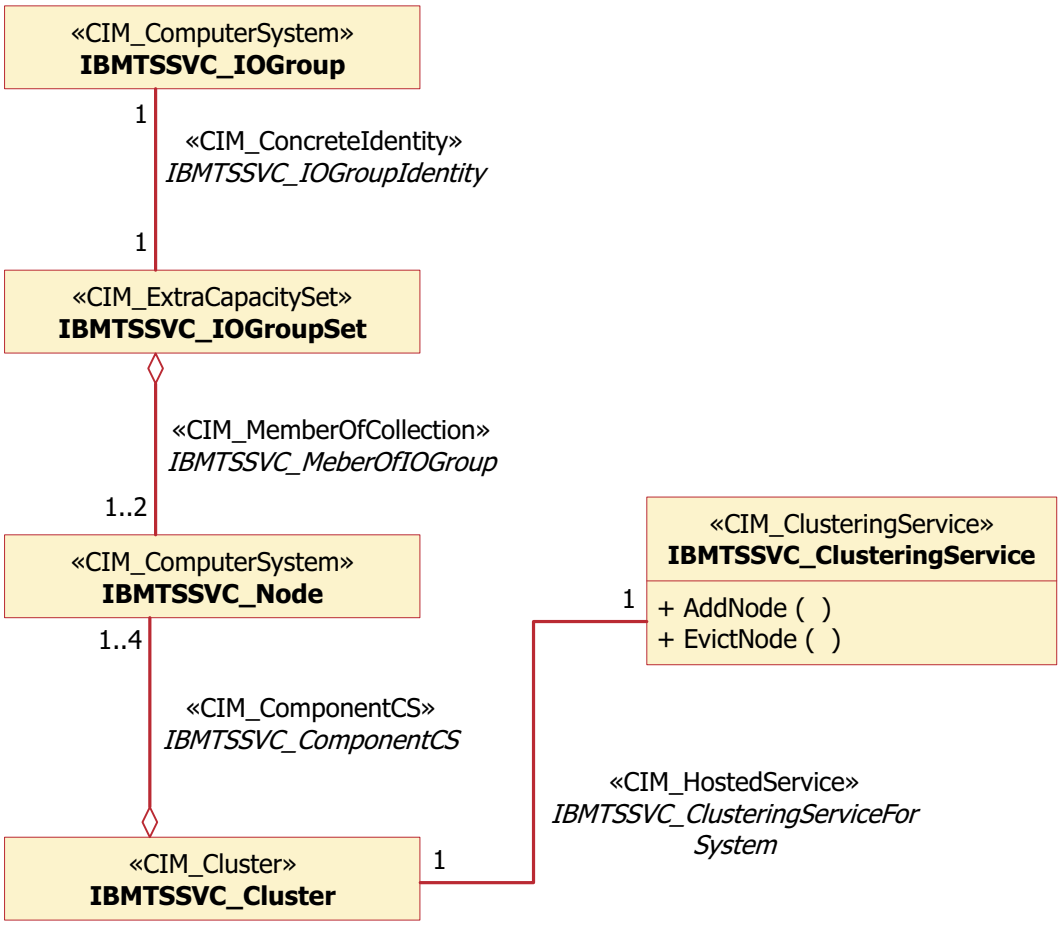

Figure 8. High-level overview of the ExtraCapacitySet subprofile of the CIM Agent for the SAN Volume Controller.

Access point subprofile

The graphic below provides a high-level overview of the access point subprofile of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 9 on page 16 shows the basic classes (building blocks) for the model.

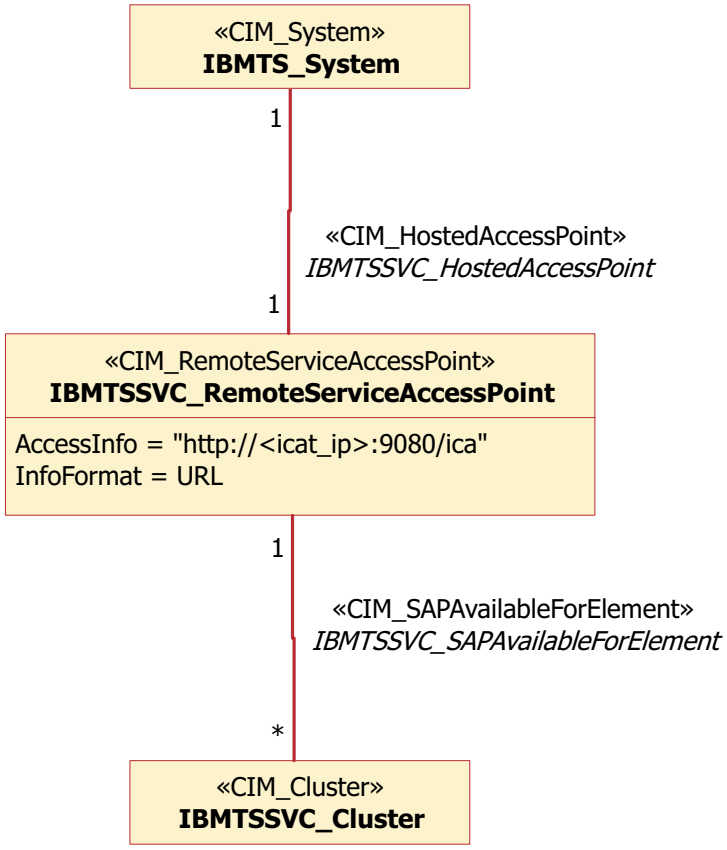

Figure 9. High-level overview of the access point subprofile of the CIM Agent for the SAN Volume Controller.

Cluster subprofile

The graphic below displays a class diagram of a clustering instance.

Figure 10 on page 17 shows the classes and associations that are important for providing the functionality of clustering service.

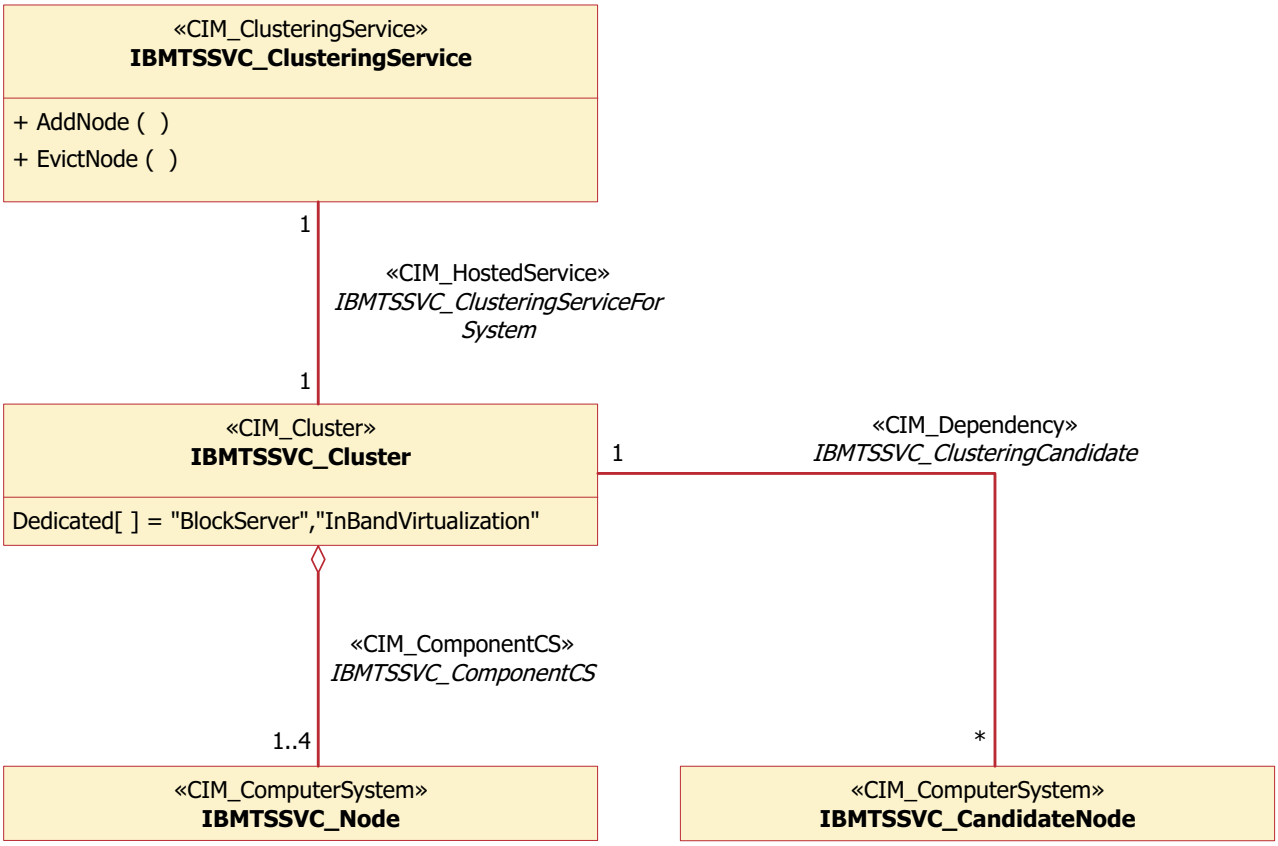

Figure 10. Class diagram of Clustering instance

Vendor-specific storage configuration operations

The graphic below provides a class diagram of a StorageConfiguration instance.

Figure 11 on page 18 shows the object classes that are important to implement basic storage configuration.

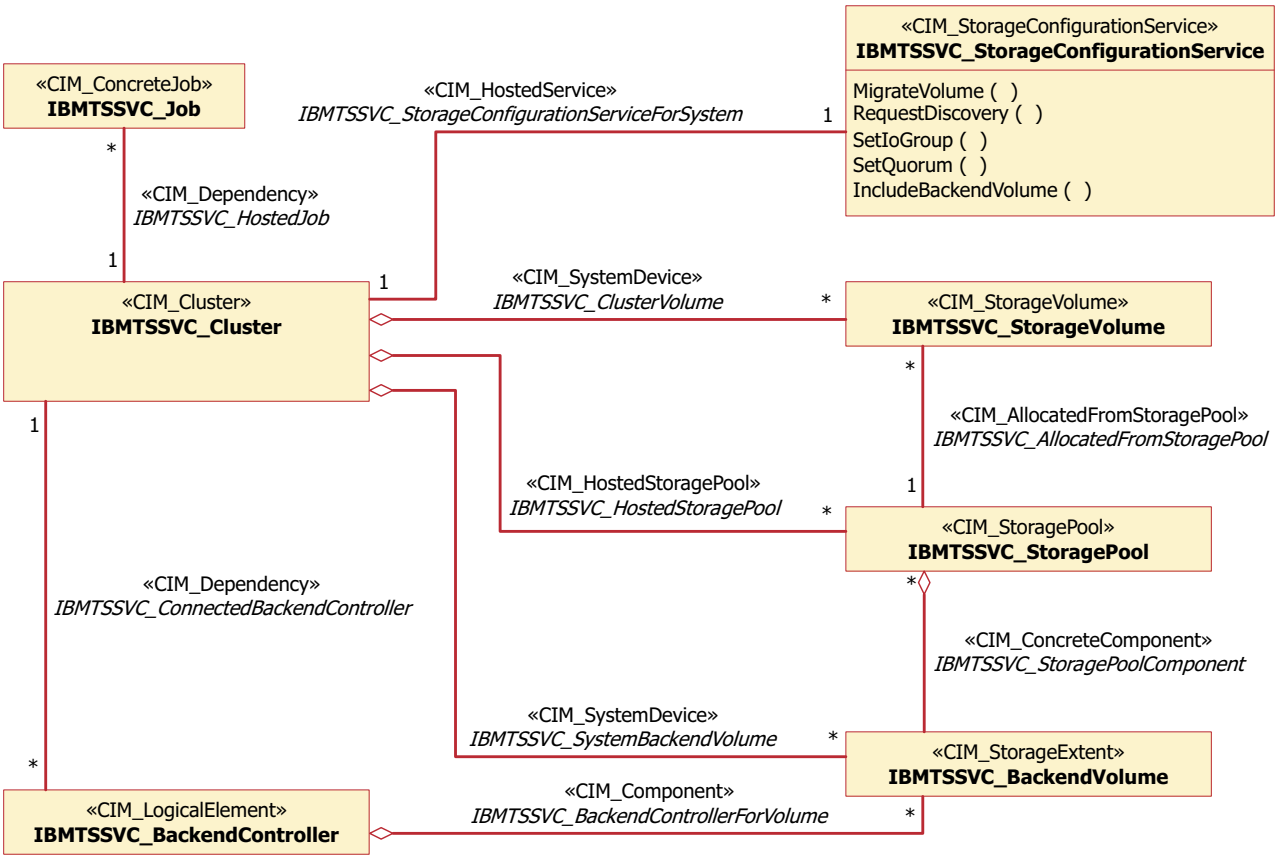

Figure 11. Class diagram of StorageConfiguration instance

LUN masking

The graphic below provides a class diagram for a LUN masking instance.

Figure 12 on page 19 shows the object classes that are important to execute LUN masking. In the CIM Agent for the SAN Volume Controller, the IBMTSSVC_StorageHardwareIdManagementService, IBMTSSVC_PrivilegeManagementService, and IBMTSSVC_ControllerConfigurationService classes provide the methods to map an IBMTSSVC_StorageVolume instance to an IBMTSSVC_HardwareIdCollection instance of an IBMTSSVC_StorageHardwareID instance.

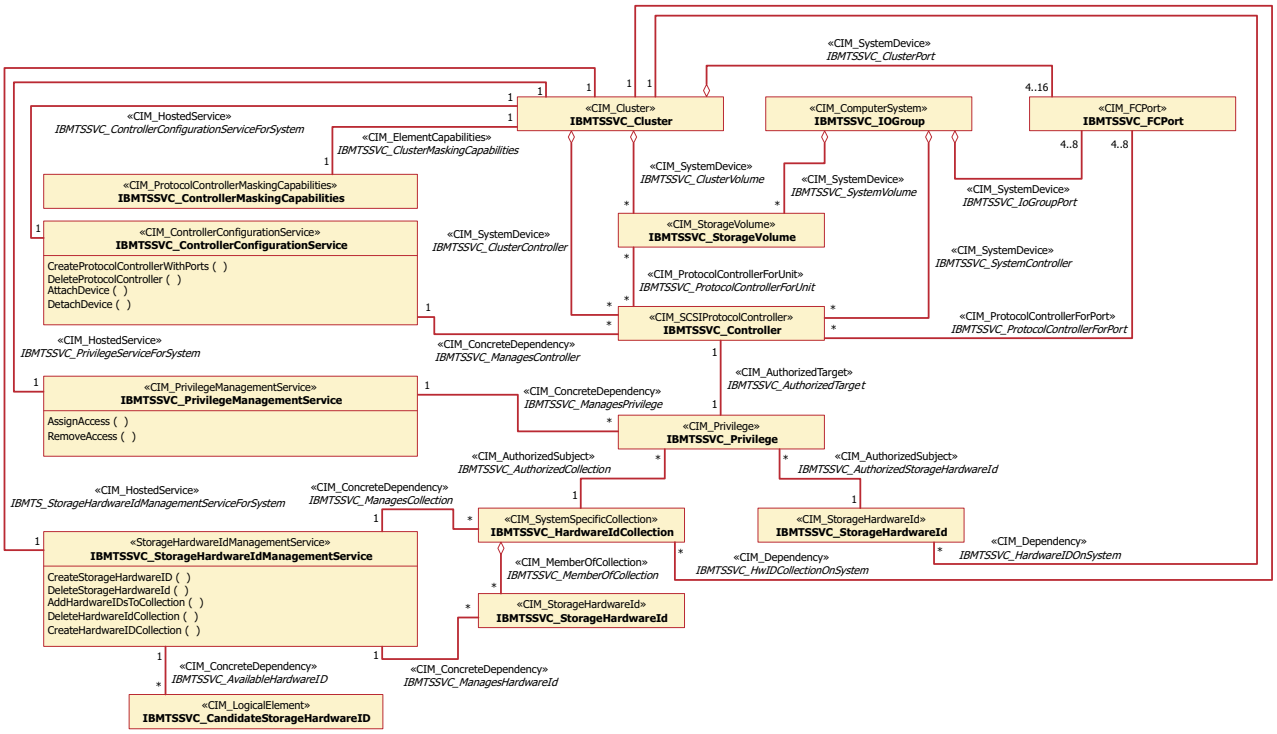

Figure 12. Class diagram for LUN masking instances

Related concepts

"LUN masking" on page 37
The SAN Volume Controller provides logical unit number (LUN) masking capability.

LUN creation subprofile

The graphic below provides a high-level overview of the LUN creation subprofile of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 13 on page 20 shows the basic classes (building blocks) for the model.

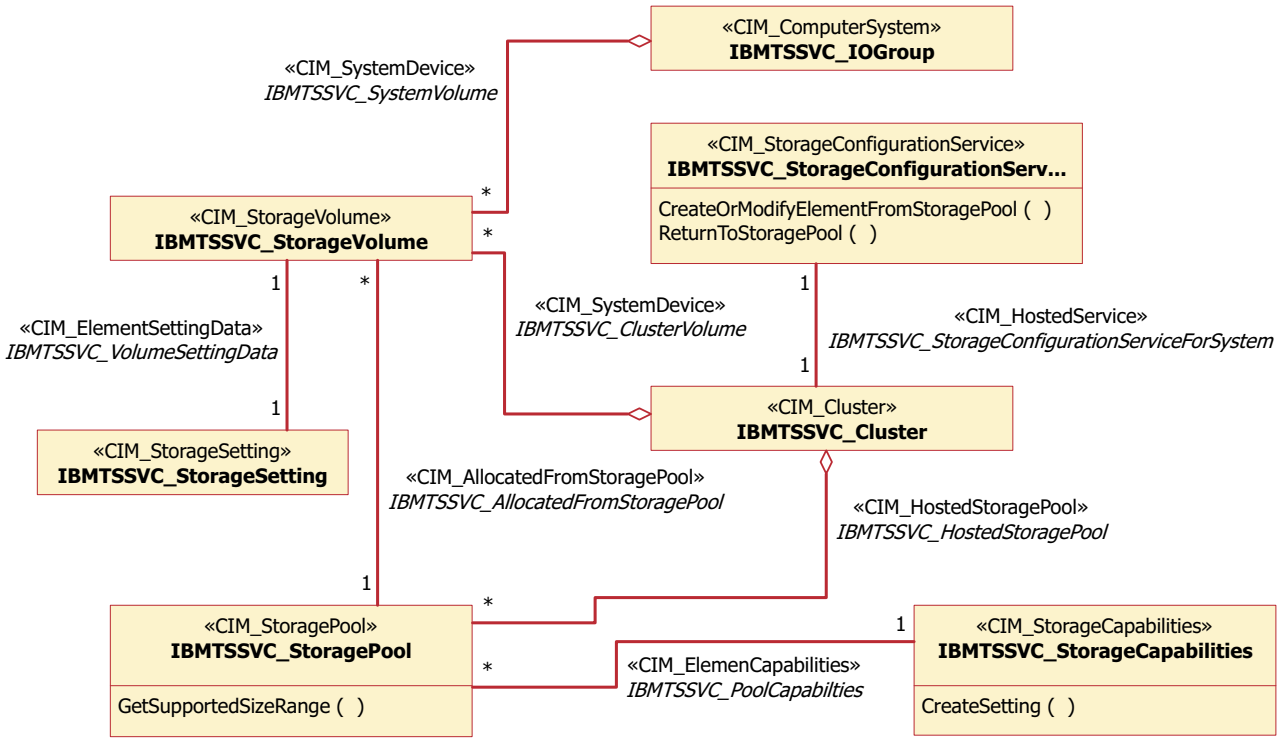

Figure 13. High-level overview of the LUN creation subprofile of the CIM Agent for the SAN Volume Controller.

Copy services

The IBMTSSVC_StorageConfigurationService class provides the methods to create copy relationships.

Figure 14 on page 21 shows the object classes that are important to provide FlashCopy, Remote Copy, and synchronous copy services. In the Common Information Model (CIM) Agent for the SAN Volume Controller, the IBMTSSVC_StorageConfigurationService class provides the methods to create copy relationships. An IBMTSSVC_StorageConfigurationService instance is always associated with an IBMTSSVC_Cluster instance.

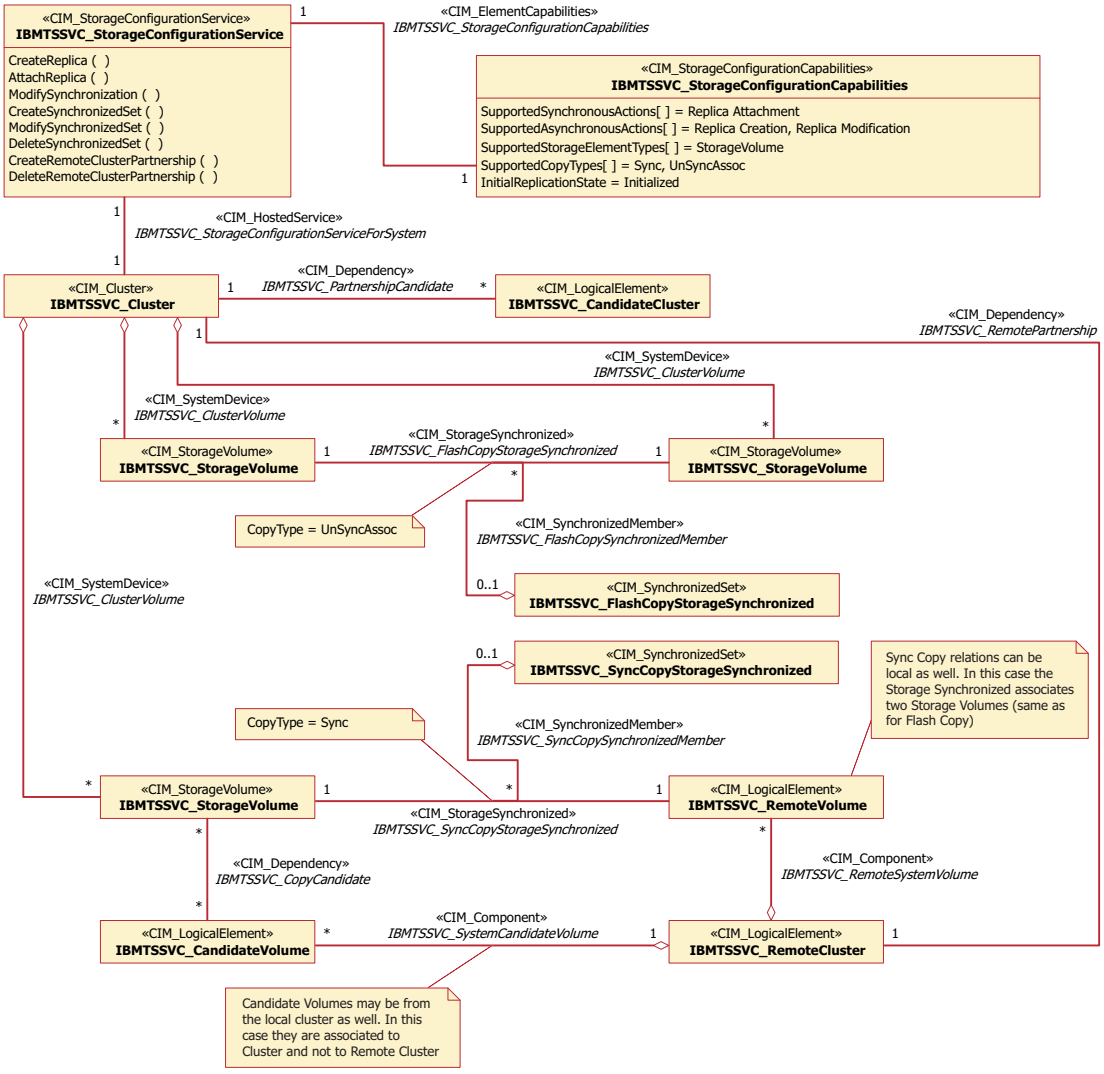

Figure 14. Class diagram of copy services instances

Vendor-specific service mode subprofile

The graphic below provides a high-level overview of the vendor-specific service mode subprofile of the Common Information Model (CIM) Agent for the SAN Volume Controller.

The service mode subprofile is supported for Blade SAN Volume Controllers (model 2062) only. Figure 15 on page 22 shows the basic classes (building blocks) for the model.

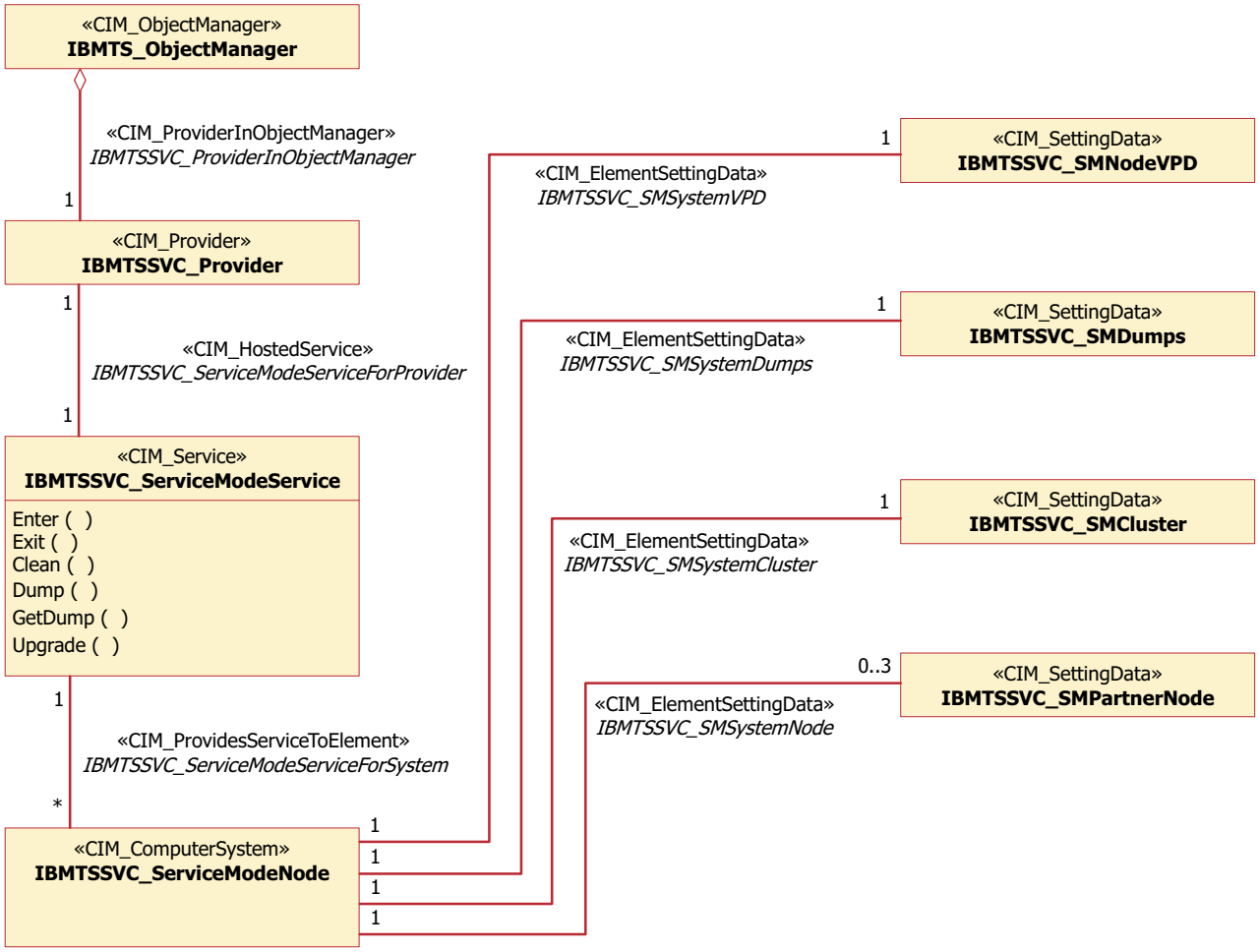

Figure 15. High-level overview of the vendor-specific service mode subprofile of the CIM Agent for the SAN Volume Controller.

Vendor-specific cluster operations

The graphic below provides a high-level overview of the vendor-specific cluster operations of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 16 on page 23 shows the basic classes (building blocks) for the model.

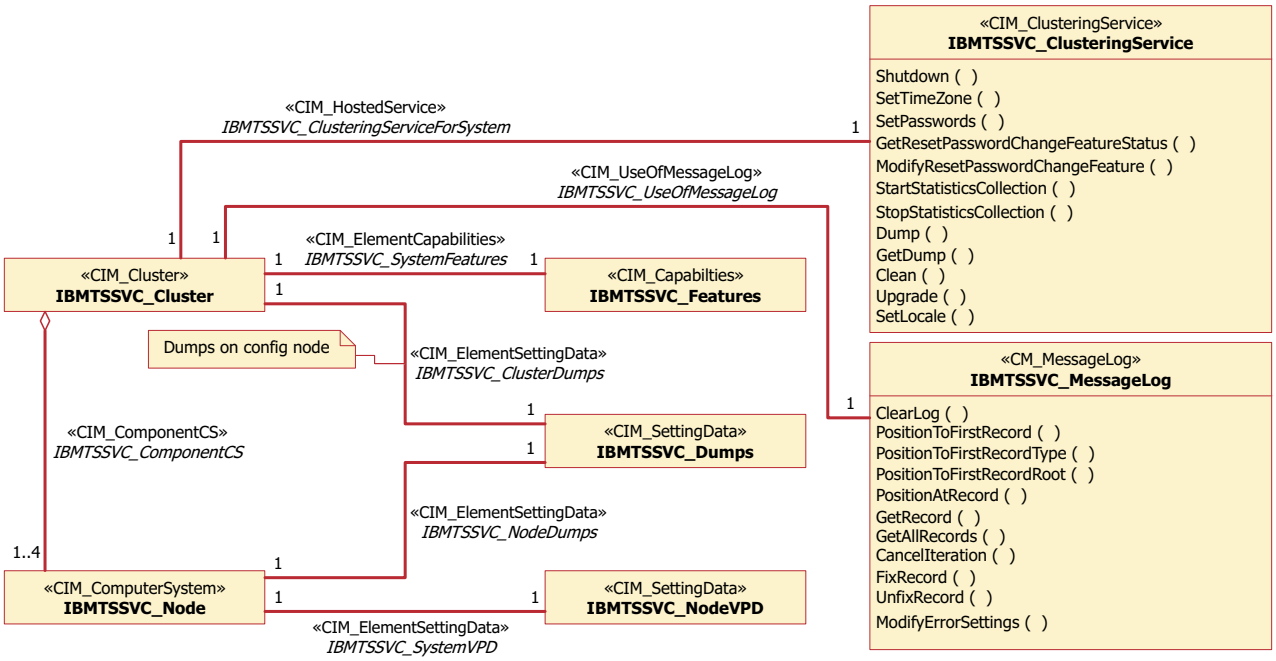

Figure 16. High-level overview of the vendor-specific cluster operations of the CIM Agent for the SAN Volume Controller.

Security service

The graphic below provides a class diagram of security instances.

Figure 17 on page 24 shows the object classes that are important to manage user accounts and their access rights on the Common Information Model Object Manager (CIMOM) and the clusters. In the Common Information Model (CIM) Agent for the , the IBMTS_AccountManagementService class provides the methods to create, delete, and modify IBMTS_Account instances, each of which represents a user. You can also use the IBMTSSVC_AccountManagementService and the IBMTSSVC_AuthorizationService classes.

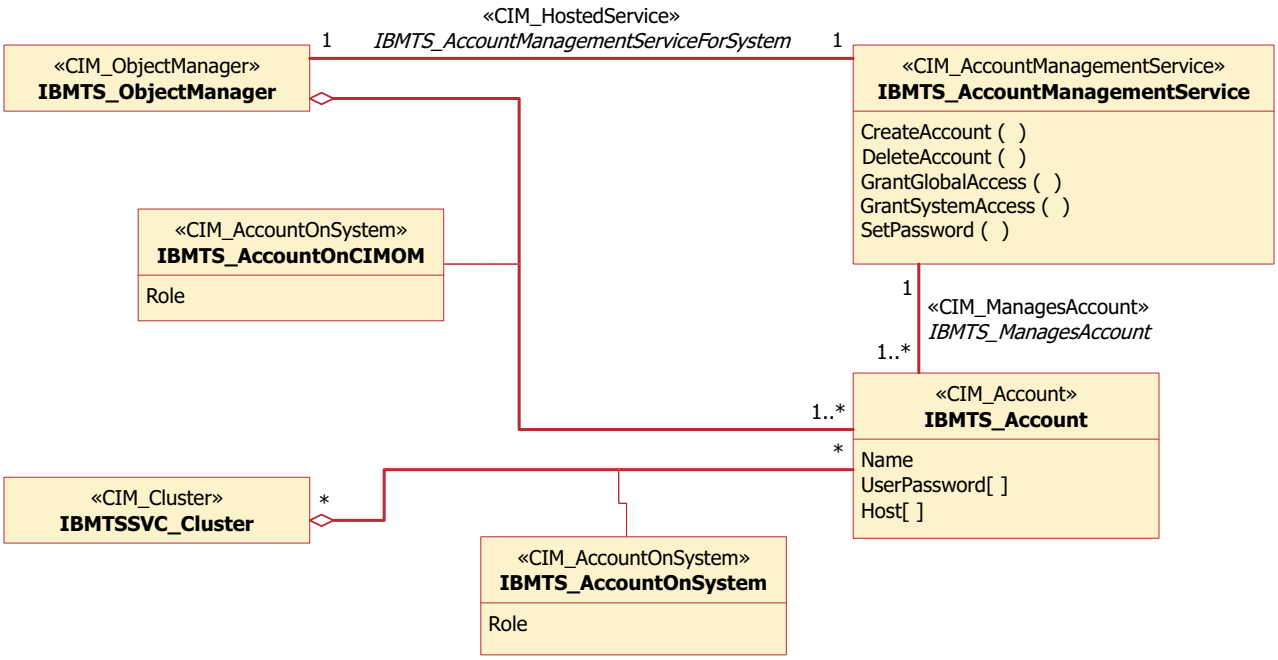

Figure 17. Class diagram of security instances

Pool Manipulation

The graphic below provides a high-level overview of pool manipulation of the Common Information Model (CIM) Agent for the SAN Volume Controller.

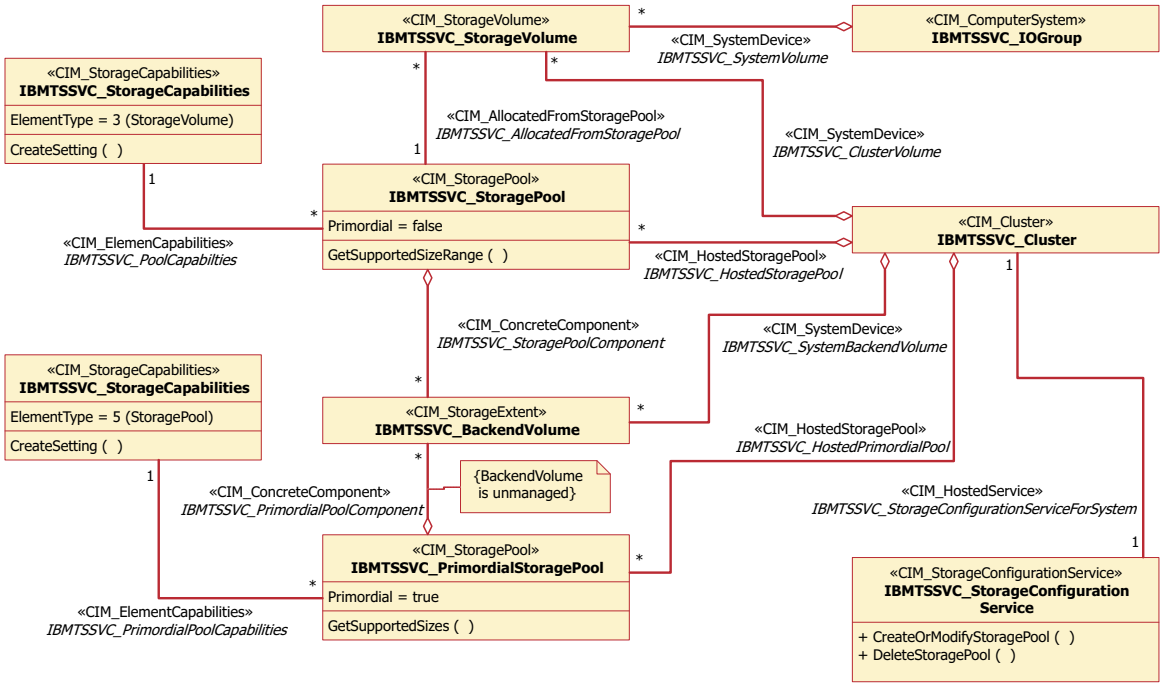

Figure 18. High-level overview of pool manipulation of the CIM Agent for the SAN Volume Controller.

Chapter 2. Performing storage configuration

The following sections describe how to use the Common Information Model (CIM) Agent object class instances to complete basic storage configuration tasks.

The following basic storage configuration tasks are discussed:

- Add a node to a cluster
- Create or modify a storage pool
- Create a storage volume

Storage configuration

Storage configuration refers to the mapping of the back-end storage to the storage pools and the allocation of volumes from the pools.

In the Common Information Model (CIM) Agent for the SAN Volume Controller, storage configuration involves three layers of objects. The objects in the back-end layer contain the back-end controllers and volumes, those in the middle layer contain the storage pools, and those in the front-end layer contain the storage volumes that are exposed to the hosts.

Performing basic storage configuration

The IBMTSSVC_StorageConfigurationService class provides the CreateOrModifyStoragePool() and CreateOrModifyElementFromStoragePool() methods for performing basic storage configuration.

You can use the CreateOrModifyStoragePool() method to create an IBMTSSVC_StoragePool and add or remove an IBMTSSVC_BackendVolume. You can use the CreateOrModifyElementFromStoragePool() method to allocate, expand, or shrink an IBMTSSVC_StorageVolume from an IBMTSSVC_StoragePool.

You must complete the initial setup of the SAN Volume Controller. That means, you have already created and added clusters to the Common Information Model Object Manager (CIMOM) configuration file of the CIM Agent, and as a result, the Common Information Model (CIM) Agent has discovered all the back-end volumes required for the storage configuration.

Perform the following tasks to complete basic storage configuration:

1. Add a node to the cluster.
2. Create a storage pool.
3. Modify the storage pool.
4. Create a storage volume.

Adding a candidate node to a cluster

You can add an IBMTSSVC_CandidateNode to an existing IBMTSSVC_Cluster.

Perform the following steps to add an IBMTSSVC_CandidateNode to an existing IBMTSSVC_Cluster:

1. Obtain the reference (CIMObjectPath) of the IBMTSSVC_Cluster to which you want to add an IBMTSSVC_CandidateNode.

© Copyright IBM Corp. 2003, 2004

2. Locate the IBMTSSVC_ClusteringService instance that is associated with the IBMTSSVC_Cluster by traversing the IBMTSSVC_ClusteringServiceForSystem association.
3. Invoke the IBMTSSVC_ClusteringService.AddNode() method while specifying the node and panel names and the references to the IBMTSSVC_CandidateNode.

Creating a new storage pool

The IBMTSSVC_StorageConfigurationService class provides the methods for creating a new IBMTSSVC_StoragePool.

You must include an IBMTSSVC_StorageConfigurationService instance in each cluster before you create a new IBMTSSVC_StoragePool instance.

Perform the following steps to create a new IBMTSSVC_StoragePool instance:

1. Obtain the reference (CIMObjectPath) of an IBMTSSVC_StorageConfigurationService instance that is associated with the IBMTSSVC_Cluster in which you will create the new storage pool by traversing the IBMTS_StorageConfigurationServiceForSystem association.
2. Invoke the IBMTSSVC_StorageConfigurationService.CreateOrModifyStoragePool method while specifying the Extent[] parameter with a list of IBMTSSVC_BackendVolume instances.

   The Extent[] parameter is a string array that contains the representation of the CIMObjectPath to an IBMTSSVC_BackendVolume.

Modifying a storage pool

You can modify an IBMTSSVC_StoragePool instance by changing the pool name and adding or removing an IBMTSSVC_BackendVolume instance from the pool.

1. Perform the following steps to modify an IBMTSSVC_StoragePool instance:
   a. Select the IBMTSSVC_StoragePool instance that you want to modify from an IBMTSSVC_Cluster.
   b. Identify the IBMTSSVC_StorageSettingPool instance that contains the parameter settings of the IBMTSSVC_StoragePool instance.
   c. Invoke the IBMTSSVC_StoragePool.SetProperty() method to change the name of the selected IBMTSSVC_StoragePool instance.
2. If necessary, you can further modify the IBMTSSVC_StoragePool by adding or removing an IBMTSSVC_BackendVolume instance to the pool:
   a. Invoke the IBMTSSVC_StorageSettingPool.CreateOrModifyStoragePool method while specifying the Extent[] parameter with information about the IBMTSSVC_BackendVolume to be added or removed from the pool. The Extent[] parameter is a string array that contains the representation of the CIMObjectPath to an IBMTSSVC_BackendVolume.

      BackendVolumes in Extent[] that are in the pool will be removed, and BackendVolumes that are not in the pool will be added. Also, you can specify the property Force of StorageSettingPool to remove BackendVolumes from the pool even if there is data on these disks. This will trigger the migration process that moves data from the removed volumes to the remaining ones.

Creating a new storage volume

The IBMTSSVC_StorageConfigurationService class provides all the methods required for creating, modifying, and deleting an IBMTSSVC_StorageVolume.

In the Common Information Model (CIM) Agent for the SAN Volume Controller, the IBMTSSVC_StorageConfigurationService class provides all the methods required for creating, modifying, and deleting an IBMTSSVC_StorageVolume.

Perform the following steps to create a new IBMTSSVC_StorageVolume instance:

1. Obtain the reference (CIMObjectPath) of the IBMTSSVC_StorageConfigurationService instance that is associated with the IBMTSSVC_Cluster to which you will assign the new volume.
2. Invoke the IBMTSSVC_StorageConfigurationService. CreateOrModifyElementFromStoragePool() method to create the new IBMTSSVC_StorageVolume with the following parameter specifications:
   a. Set ElementType to 2.
   b. Set Size to the desired volume size in megabytes.
   c. Obtain the reference (CIMObjectPath) of the IBMTSSVC_StoragePool instance from which you will allocate an IBMTSSVC_StorageVolume.
   d. Set InPool to the reference (obtained in the previous step) of the pool from which the volume will be allocated.

Chapter 3. Performing Copy Services

This chapter describes how you can use the Common Information Model Agent object class instances to establish new Copy Services relationships.

It provides step-by-step procedures for you to perform the following Copy Services tasks:

1. "Creating a new FlashCopy relationship between storage volumes"
2. "Creating a FlashCopy relationship for a synchronized set" on page 32
3. "Creating a synchronous copy relationship between volumes in the same cluster" on page 33
4. "Creating a synchronous copy relationship between volumes in different clusters" on page 33

Copy Services

FlashCopy and synchronous Remote Copy are the copy services provided by the SAN Volume Controller.

These copy services are available to all supported hosts that are connected to the SAN Volume Controller.

The FlashCopy service enables you to make an instant, point-in-time copy of a source IBMTSSVC_StorageVolume to a target IBMTSSVC_StorageVolume. The synchronous copy service provides a consistent copy of the source IBMTSSVC_StorageVolume on the target IBMTSSVC_StorageVolume. Data is written to the target volume synchronously after it is written to the source volume, both of which can belong to the same IBMTSSVC_Cluster or different IBMTSSVC_Clusters.

Creating a new FlashCopy relationship between storage volumes

The IBMTSSVC_StorageConfigurationService class provides the methods for establishing a FlashCopy relationship between two IBMTSSVC_StorageVolume instances that are the same size and belong to the same IBMTSSVC_Cluster.

Perform the following steps to create a FlashCopy relationship between two IBMTSSVC_StorageVolume instances:

1. Select an IBMTSSVC_StorageVolume instance as the source volume for the desired FlashCopy relationship.
2. Select an associated IBMTSSVC_CandidateVolume instance as the target volume.

   Make sure that the source IBMTSSVC_StorageVolume and target IBMTSSVC_CandidateVolume instances belong to the same IBMTSSVC_Cluster.
3. Retrieve the IBMTSSVC_StorageConfigurationService instance associated with the IBMTSSVC_Cluster to which the selected IBMTSSVC_StorageVolume instances belong.
4. Invoke the IBMTSSVC_StorageConfigurationService.AttachReplica() method with the following parameter specifications:
   a. Set SourceElement to the reference (CIMObjectPath) of the source IBMTSSVC_StorageVolume.

© Copyright IBM Corp. 2003, 2004

b. Set TargetElement to the reference (CIMObjectPath) of the target IBMTSSVC_CandidateVolume.
c. Optionally set ElementName to the name of the synchronization.
d. Optionally set BackgroundCopyRate to the desired priority of the background copy rate in percent (0-100%).
e. Optionally specify Set to add the newly created FlashCopySynchronization to the set. If you specify a null value, the newly created FlashCopySynchronization will not be a member of a synchronized set.
f. Set CopyType to 4.

The source IBMTSSVC_StorageVolume and target IBMTSSVC_CandidateVolume instances are now connected through the IBMTSSVC_FlashCopyStorageSynchronized association.

Creating a FlashCopy relationship for a synchronized set

The IBMTSSVC_StorageConfigurationService class provides the methods for establishing a FlashCopy relationship between two IBMTSSVC_StorageVolume instances and then adding it to an IBMTSSVC_FlashCopySynchronizedSet.

Perform the following steps to create a FlashCopy relationship between two IBMTSSVC_StorageVolume instances and add it to an IBMTSSVC_FlashCopySynchronizedSet:

1. Select an IBMTSSVC_StorageVolume instance as the source volume for the desired FlashCopy relationship.
2. Select an associated IBMTSSVC_CandidateVolume instance as the target volume.

   Make sure that the source IBMTSSVC_StorageVolume and target IBMTSSVC_CandidateVolume instances are the same size.
3. Retrieve the IBMTSSVC_StorageConfigurationService instance associated with the IBMTSSVC_Cluster to which the selected IBMTSSVC_StorageVolume instances belong.
4. Invoke the IBMTSSVC_StorageConfigurationService.AttachReplica() method with the following parameter specifications:
   a. Set SourceElement to the reference (CIMObjectPath) of the source IBMTSSVC_StorageVolume.
   b. Set TargetElement to the reference (CIMObjectPath) of the target IBMTSSVC_CandidateVolume.
   c. Optionally set ElementName to the name of the synchronization.
   d. Optionally set BackgroundCopyRate to the desired priority of the background copy rate in percent (0-100%).
   e. Set CopyType to 4.

   The source IBMTSSVC_StorageVolume and target IBMTSSVC_CandidateVolume instances are now connected through the IBMTSSVC_FlashCopyStorageSynchronized association.
5. Create an IBMTSSVC_FlashCopySynchronizedSet instance by invoking the IBMTSSVC_StorageConfigurationService.CreateSynchronizedSet() method with the following parameter specifications:
   a. Set CopyType to 4 (flash).
   b. Optionally set ElementName to the name of the newly created IBMTSSVC_FlashCopySynchronizedSet.

6. Add the IBMTSSVC_FlashCopyStorageSynchronized instance to the IBMTSSVC_FlashCopySynchronizedSet by invoking the IBMTSSVC_StorageConfigurationService.ModifySynchronizedSet() method with the Operation parameter set to 0 (add).

   The synchronization must belong to the same cluster as the hosting service.

Creating a synchronous copy relationship between volumes in the same cluster

The IBMTSSVC_StorageConfigurationService class provides the methods for creating a synchronous copy relationship between a source IBMTSSVC_StorageVolume and a target IBMTSSVC_StorageVolume or a source IBMTSSVC_StorageVolume and a target IBMTSSVC_CandidateVolume in the same IBMTSSVC_Cluster.

Perform the following steps to create the synchronous copy relationship:

1. Select an IBMTSSVC_StorageVolume instance as the source volume for the desired synchronous copy relationship.
2. Select an IBMTSSVC_StorageVolume instance or an IBMTSSVC_CandidateVolume instance as the target volume.
3. Obtain the reference (CIMObjectPath) of the IBMTSSVC_StorageConfigurationService instance associated with the IBMTSSVC_Cluster to which the selected volumes belong.
4. Invoke the IBMTSSVC_StorageConfigurationService.AttachReplica() method with the following parameter specifications:
   a. Set SourceElement to the reference (CIMObjectPath) of the source IBMTSSVC_StorageVolume instance.
   b. Set TargetElement to the reference (CIMObjectPath) of the target IBMTSSVC_StorageVolume or IBMTSSVC_CandidateVolume instance.
   c. Optionally set ElementName to the name of the synchronization.
   d. Set CopyType to 3.

The source IBMTSSVC_StorageVolume instance and the target IBMTSSVC_StorageVolume or IBMTSSVC_CandidateVolume instance (whichever you selected) are now connected through the IBMTSSVC_SyncCopyStorageSynchronized association.

Creating a synchronous copy relationship between volumes in different clusters

The IBMTSSVC_StorageConfigurationService class provides the methods for creating a synchronous copy relationship between a source IBMTSSVC_StorageVolume instance and a target IBMTSSVC_CandidateVolume instance belonging to different IBMTSSVC_Cluster instances.

Perform the following steps to create a synchronous copy relationship between two volumes with the source located in a local cluster and the target located in a remote cluster:

1. Identify an IBMTSSVC_Cluster as the source cluster for the desired synchronous copy relationship.

2. Obtain the reference (CIMObjectPath) of the IBMTSSVC_StorageConfigurationService instance associated with the source cluster.
3. Identify the IBMTSSVC_CandidateCluster on which you want the synchronous copy to reside by traversing the IBMTSSVC_ClusterScopeRemoteCluster association.
4. Invoke the IBMTSSVC_StorageConfigurationService.CreateRemoteClusterPartnerShip() method with the following parameter specifications:
   a. Set RemoteCluster to the reference (CIMObjectPath) of the IBMTSSVC_CandidateCluster.
   b. Optionally set Bandwidth to the desired bandwidth in megabytes (MB).

   Make sure to issue the method from both the source and candidate clusters to establish a fully configured partnership; otherwise, the synchronous copy relationship cannot be established.
5. Select an IBMTSSVC_StorageVolume as the source volume from the source IBMTSSVC_Cluster.
6. Select an IBMTSSVC_CandidateVolume as the target volume from the IBMTSSVC_RemoteCluster. (IBMTSSVC_StorageVolumes on the remote cluster are seen on the local cluster as IBMTSSVC_CandidateVolumes).
7. Invoke the IBMTSSVC_StorageConfigurationService.AttachReplica() method with the following parameter specifications:
   a. Set SourceElement to the reference (CIMObjectPath) of the source IBMTSSVC_StorageVolume instance.
   b. Set TargetElement to the reference (CIMObjectPath) of the target IBMTSSVC_StorageVolume or IBMTSSVC_CandidateVolume instance.
   c. Optionally set ElementName to the name of the synchronization.
   d. Set CopyType to 3.

The source IBMTSSVC_StorageVolume instance and the target IBMTSSVC_StorageVolume or IBMTSSVC_CandidateVolume instance (whichever you selected) are now connected through the IBMTSSVC_SyncCopyStorageSynchronized association.

Flash Copy state diagram

The graphic below provides a high-level overview of the Flash Copy state diagram of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 19 on page 35 shows the supported states and transitions for Flash Copy relationships.

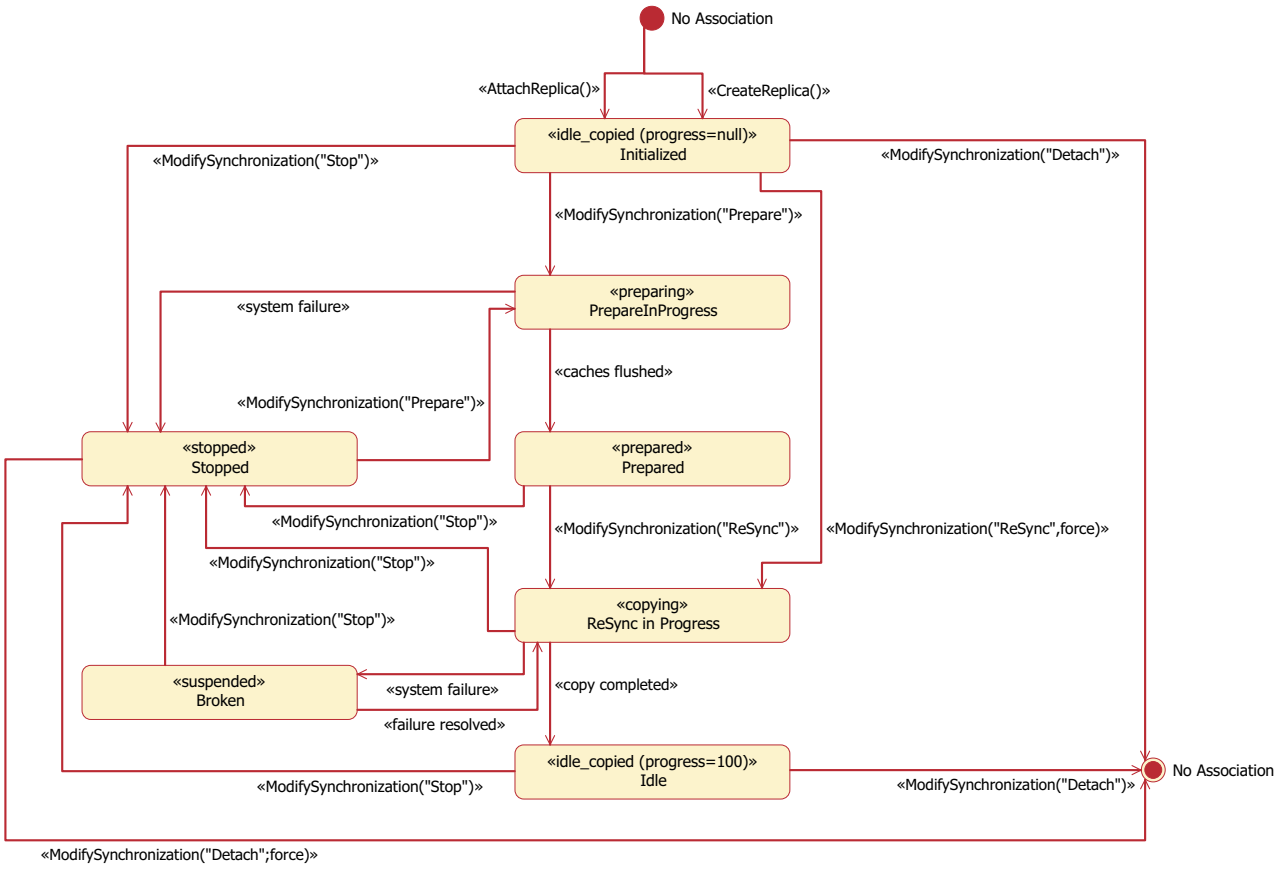

Figure 19. Flash Copy state diagram of the CIM Agent for the SAN Volume Controller.

Sync Copy state diagram

The graphic below provides a high-level overview of the Sync Copy state diagram of the Common Information Model (CIM) Agent for the SAN Volume Controller.

Figure 20 on page 36 shows the supported states and transitions for Flash Copy relationships.

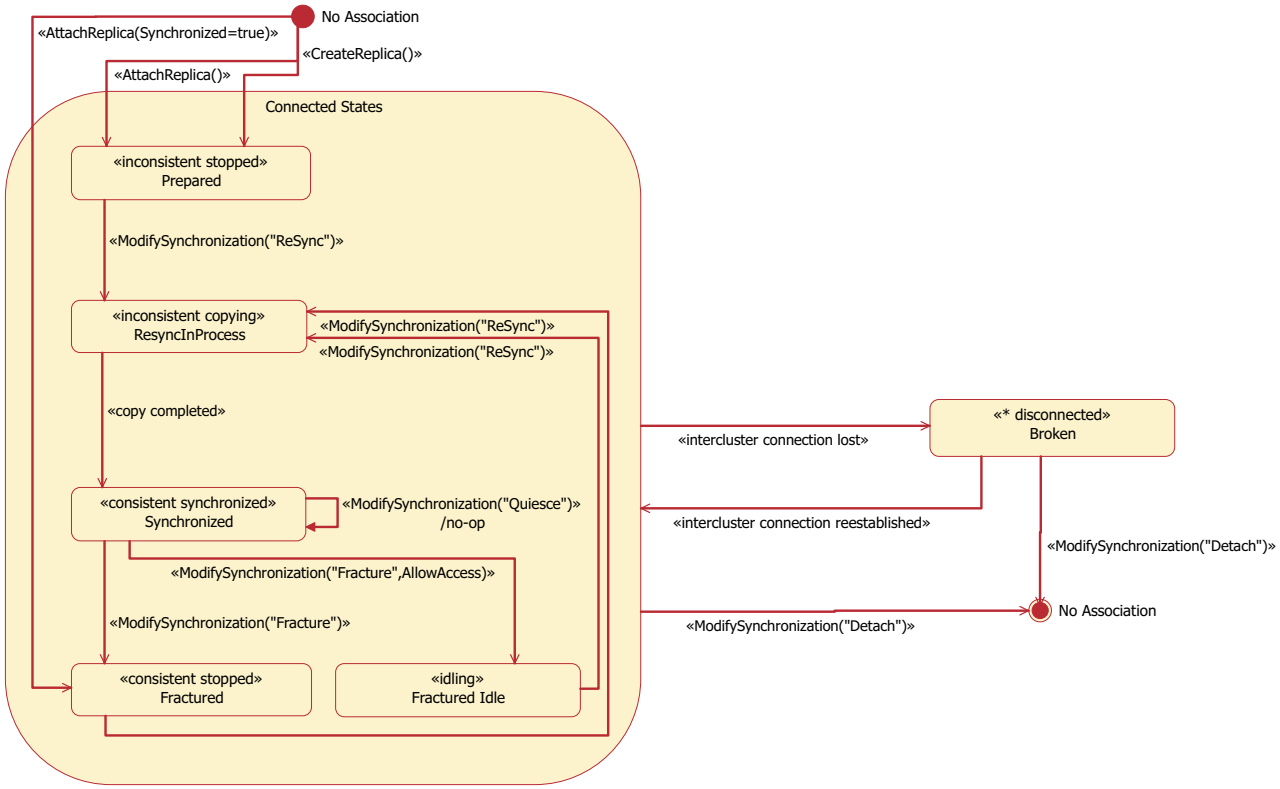

Figure 20. High-level overview of the Sync Copy state diagram of the CIM Agent for the SAN Volume Controller.

Chapter 4. Performing LUN masking

This chapter describes how you can use the CIM Agent object class instances to perform LUN masking.

LUN masking

The SAN Volume Controller provides logical unit number (LUN) masking capability.

This allows you to associate or dissociate a storage volume with a fibre-based host initiator through the worldwide port numbers (WWPNs).

Performing LUN masking

To perform logical unit number (LUN) masking, you first need a volume to map, which is represented by an instance of IBMTSSVC_StorageVolume.

You may map host ports as well as entire hosts to a volume:

- Host = IBMTSSVC_HarwareIdCollection
- Port = IBMTSSVC_StorageHardwareID

IBMTSSVC_HarwareIdCollection aggregates instances of IBMTSSVC_StorageHardwareID.
Before performing LUN masking, you need instances of the following services (the cluster is the referenced object). Both are available within a cluster scope (IBMTSSVC_Cluster).

- IBMTSSVC_ControllerConfigurationService (association IBMTSSVC_ControllerConfigurationServiceForSystem)
- IBMTSSVC_StorageHardwareIdManagementService (association IBMTSSVC_StorageHardwareIdManagementServiceForSystem)

1. Select an IBMTSSVC_StorageVolume (LUN) instance and either an IBMTSSVC_HardwareIdCollection (Host) instance or an IBMTSSVC_StorageHardwareID (Port) instance (these are associated with each other). The association between the two instances is IBMTSSVC_MemberOfCollection.

   Note: If you wish to work with the host and host ports, obtain the reference of IBMTSSVC_StorageHardwareIdManagementService by traversing either IBMTSSVC_ManagesCollection or IBMTSSVC_ManagesHardwareID.
2. Obtain the reference (CIMObjectPath) of both instances.
3. Traverse the IBMTSSVC_SystemVolume association from the StorageVolume to obtain the reference of the scoping IBMTSSVC_IOGroup.
4. If starting from IBMTSSVC_StorageHardwareID, optionally traverse IBMTSSVC_HardwareIdCollection to get the IBMTSSVC_HardwareIdCollection. This is necessary in order to check the associated instance of IBMTSSVC_Privilege in a later step. It is also possible to directly traverse the association IBMTSSVC_AuthorizedStorageHardwareId to get to the associated instance of IBMTSSVC_Privilege.
5. Check to see if a controller already exists for the Host. This can be done by traversing the IBMTSSVC_AuthorizedCollection to obtain the reference of the IBMTSVC_Privilege instance. If starting from IBMTSSVC_StorageHardwareID, you can traverse the IBMTSSVC_AuthorizedStorageHardwareID to obtain the

© Copyright IBM Corp. 2003, 2004

reference of the IBMTSVC_Privilege instance. If a controller does not already exist for the Host, a new controller needs to be created using the ControllerConfigurationService. A privilege has a 1:1 relation to the controller.

6. If a privilege is already associated, obtain the reference of the IBMTSSVC_Controller instance by traversing the association: IBMTSSVC_AuthorizedTarget. If there is no controller, see 11.
7. Check to make sure that the IOGroup of the volume (LUN) and the IOGroup of the controller are the same.
8. Traverse the IBMTSSVC_SystemController association to obtain the reference of the scoping IOGroup.
9. Select the Controller which belongs to the same IOGroup as the StorageVolume.
10. Call AttachDevice() on that Controller instance. Make sure to set the Device parameter to the reference of the StorageVolume.
11. If no controller is available, use the IBMTSSVC_ControllerConfigurationService to create one. This class offers you a method called ″CreateProtocolControllerWithPorts().″ In order to invoke this method, you need the FCPorts and the Host. For Details see HLD Version 1.6 chapter 6.11.1. A more comfortable way to use this method is to use the reference of the IOGroup rather than the FCPorts. The methods automatically detects the belonging FCPorts.

Related reference

“IBMTSSVC_StorageHardwareIDManagementService” on page 202
This service provides extrinsic methods to manage HardwareAccounts and Hosts for the SAN Volume Controller.

Chapter 5. Network Considerations

This chapter describes the two methods ICAT uses to publish its services. The CIM Agent service can be published through SLP based discovery. The User Interface connection information is published by an instance of the RemoteServiceAccessPoint provided by the CIM Agent.

SLP based discovery

The CIMOM automatically registers its IP address with the SLP; however, registration can be modified manually.

In an environment with multiple network adapters, the SLP Service Agent may register the CIM Agent with the IP of a network adapter in a different subnet than the management application. As a result, the management application cannot discover the CIM Agent.

The example below illustrates why this occurs:

- The management application runs in subnet A.
- The CIM Agent machine has adapters for subnet A and subnet B.
- Using SLP based discovery, the CIM Agent was registered for subnet B.
- When the management application runs discovery, it detects the CIM Agent SLP registration with the IP of subnet B.
- The management application's connection attempt to the CIM Agent fails because the IP of subnet B cannot be reached from subnet A.

To correct this issue, manually register the IP with the SLP. This can be done by setting an attribute within the **cimom.properties** file.

Perform the following steps to set the attribute:

1. Go to the CIM Agent installation directory.
2. Open the cimom.properties file.
3. Find the attribute: iPToRegisterWithSLP=Off.

   With this attribute set to off, the CIM Agent automatically detects the primary network card.
4. Set iPToRegisterWithSLP=<Network Adapter's IP Address>.

   This will change the IP address to the correct network adapter, allowing the management application to discover the CIM Agent.

RemoteServiceAccessPoint

In an environment with multiple network cards, it may be necessary to manually set the connection data of the RemoteServiceAccessPoint.

The RemoteServiceAccessPoint (RSAP) class hosts the information necessary for connection to the Web User Interface. Management applications can obtain an instance of the RSAP from the CIMOM to launch the User Interface through the Web.

You can manually set the connection data of the RSAP. This is helpful in an environment with multiple network cards.

© Copyright IBM Corp. 2003, 2004

Perform the following steps to set the connection data:

1. Obtain the instance of IBMSVC_Cluster.
2. Using the intrinsic method setProperty( ), modify the properties ConsoleIP and ConsolePort.

   The CIM Agent then automatically updates the RSAP.

Chapter 6. CIM Agent object classes

Object classes are the building blocks of the CIM Agent and provide functionality such as storage configuration, Copy Services, and LUN masking.

Core object classes

The following section describes the CIM Agent's core classes and their properties for the SAN Volume Controller.

IBMTSSVC_BackendController

The IBMTSSVC_BackendController class extends the CIM_SCSIController class.

Properties

The IBMTSSVC_BackendController class represents a RAID adapter on the back end of the SAN Volume Controller. The adapters control the IBMTSSVC_BackendVolumes that the SAN Volume Controller uses to store data. The IBMTSSVC_BackendController class extends the CIM_SCSIController class and has the properties shown in Table 5.

Table 5. IBMTSSVC_BackendController properties

Property	Type	Qualifier	Description
AccessGranted	Boolean	ModelCorrespondence (CIM_Controller. AuthorizationView)	Unsupported property

© Copyright IBM Corp. 2003, 2004

Table 5. IBMTSSVC_BackendController properties (continued)

Property	Type	Qualifier	Description
AdditionalAvailability	Uint16[]	Deprecated(CIM_Associated PowerManagementService. PowerState CIM_Managed SystemElement. OperationalStatus CIM_Enabled LogicalElement.EnabledStatus) ModelCorrespondence (CIM_LogicalDevice.Availability)	Additional availability and status of the Device, beyond that specified in the Availability property. **Code** **Semantics** **1** Other **2** Unknown **3** Running/ Full Power **4** Warning **5** In Test **6** N/A **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced

Table 5. IBMTSSVC_BackendController properties (continued)

Property	Type	Qualifier	Description
Availability	Uint16	Deprecated(CIM_Associated PowerManagementService. PowerState CIM_Managed SystemElement.Operational Status CIM_EnabledLogical Element.EnabledStatus), ModelCorrespondence (CIM_LogicalDevice. AdditionalAvailability), Expensive(TRUE)	The primary availability and status of the Device. **Code** **Semantics** **1** Other **2** Unknown **3** Running/ Full Power **4** Warning **5** In Test **6** N/A **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced
Caption	String	MaxLen(64)	Unsupported property
Controlled	Boolean		Unsupported property
CreationClassName	String	MaxLen(256)	Indicates the name of the class or subclass used to create an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
DeviceID	String	MaxLen(64)	An address or other identifying information to uniquely name the LogicalDevice
Description	String		Provides a textual description of the object
Element Name	String		Unsupported property

Table 5. IBMTSSVC_BackendController properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** N/A **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** N/A **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
ErrorCleared	Boolean	Deprecated (CIM_ManagedSystem Element)	Unsupported Property
ErrorDescription	String	Deprecated (CIM_DeviceErrorData. ErrorDescription)	Unsupported Property

Table 5. IBMTSSVC_BackendController properties (continued)

Property	Type	Qualifier	Description
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice. OtherIdentifyingInfo)	An array of free-form strings providing explanations and details behind the entries in the OtherIdentifyingInfo array. **Note:** Each entry of this array is related to the entry in OtherIdentifyingInfo that is located at the same index.
InstallDate	Date-time		Unsupported Property
LastErrorCode	Uint32	Deprecated(CIM_DeviceError Data.LastErrorCode)	Unsupported Property
MaxQuiesceTime	Uint64	Deprecated(No value), Units(MilliSeconds)	Unsupported Property
Name	String	MaxLen(1024), Write(TRUE), WriteRole(Administrator)	Defines the unique label, in the context of the hosting system, by which the Controller is known
Operational Status	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystemElement. StatusDescriptions), Expensive(TRUE)	Indicates the current status of the element. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode

Table 5. IBMTSSVC_BackendController properties (continued)

Property	Type	Qualifier	Description
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
OtherIdentifying Info	String[]	MaxLen(256), ArrayType (Indexed), ModelCorrespondence (CIM_LogicalDevice. IdentifyingDescriptions)	OtherIdentifying Info captures additional data, beyond DeviceID information, that could be used to identify a LogicalDevice. One example would be to hold the OperatingSystem's user-friendly name for the Device in this property.
PowerManagement Capabilities	Uint16[]	Deprecated(CIM_Power ManagementCapabilities. PowerCapabilities)	An enumerated array describing the power management capabilities of the Device. The use of this property has been deprecated. Instead, the PowerCapabilites property in an associated PowerManagement Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
PowerManagement Supported	Boolean	Deprecated(CIM_Power ManagementCapabilities)	Indicates that the device can be power-managed
PowerOnHours	Uint64	Deprecated(CIM_Powered StatisticalData.PowerOn Hours), Units(Hours), Counter(TRUE)	Unsupported Property

Table 5. IBMTSSVC_BackendController properties (continued)

Property	Type	Qualifier	Description
ProductIdHigh	String		The higher part of the controller's product id.
ProductIdLow	String		The lower part of the controller's product id.
ProductRevision	String	Expensive(TRUE)	The controller's product revision
ProductSerial Number	String		The controller's product serial number
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	An integer enumeration indicator. This property is provided to compare Requested and current Enabledstatus. Note that SAN Volume Controller does not evaluate this attribute and, therefore, no action is taken when it is changed. **Code Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystemElement)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystemElement. OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).

Table 5. IBMTSSVC_BackendController properties (continued)

Property	Type	Qualifier	Description
StatusInfo	Uint16	Deprecated (CIM_EnabledLogical Element.EnabledState)	Indicates the state (enabled or disabled) of the Logical Device. Since this property does not apply to the Controller, the value ″Not Applicable″ is always reported. **Code** **Semantics** **1** Other **2** Unknown **3** Enabled **4** Disabled **5** Not Applicable
SystemCreation ClassName	String	Propagated (CIM_System.Creation ClassName), MaxLen(256)	The scoping system's CreationClassName
SystemName	String	Propagated(CIM_System. Name), MaxLen(256)	The scoping system's name
TimeOfLastState Change	Date-time		Unsupported Property
TotalPowerOnHours	Uint64	Deprecated (CIM_PoweredStatisticalData. TotalPowerOnHours), Units (Hours), Counter(TRUE)	Unsupported Property
VendorID	String		Indicates the ID of the controller's vendor
VolumeLinkCount	Uint32	Expensive(TRUE)	Specifies the number of links to the BackendVolumes
VolumeMax LinkCount	Uint32	Expensive(TRUE)	Specifies the maximum number of links to the BackendVolumes
WWNN	String	Expensive(TRUE)	Indicates the worldwide network name (WWNN) of the controller
Wwpn	String[]	ModelCorrespondence (IBMTSSVC_BackendController. WwpnPathCount), Expensive(TRUE)	Indicates the WWPN of the controller
WwpnMax PathCount	Uint64[]	ModelCorrespondence (IBMTSSVC_BackendController. Wwpn), Expensive(TRUE)	Specifies the maximum path count to the corresponding WWPN
WwpnPathCount	Uint64[]	ModelCorrespondence (IBMTSSVC_BackendController. Wwpn), Expensive(TRUE)	Specifies the path count to the corresponding WWPN

IBMTSSVC_BackendVolume

The IBMTSSVC_BackendVolume class represents a SCSI LUN that a storage controller in the fibre-channel SAN exposes to the SAN Volume Controller.

Properties

The IBMTSSVC_BackendVolume class extends the CIM_StorageExtent class and has the properties shown in Table 6.

Table 6. IBMTSSVC_BackendVolume properties

Property	Type	Qualifier	Description
Access	Uint16		Specifies the access levels, as defined below: **Code** **Semantics** **0** Unknown **1** Readable **2** Writeable **3** Read/Write Supported **4** Write Once

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
Additional Availability	Uint16[]	ModelCorrespondence (CIM_LogicalDevice.Availability)	Specifies availability and status of the device in addition to what is specified in the Availability property. The Availability property denotes the primary status and availability of the device. In cases where this will not be sufficient to denote the complete status of the device, the AdditionalAvailability property provides further information. **Code** **Semantics** **1** Other **2** Unknown **3** Running/Full Power **4** Warning **5** In Test **6** Not Applicable **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
Availability	Uint16	Deprecated(CIM_Associated PowerManagementService. PowerState CIM_Managed SystemElement.Operational Status CIM_EnabledLogical Element.EnabledState) ModelCorrespondence (CIM_LogicalDevice. AdditionalAvailability)	Specifies the primary availability and status of the device. **Code** **Semantics** **1** Other **2** Unknown **3** Running/Full Power **4** Warning **5** In Test **6** Not Applicable **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced
BlockSize	Uint64	Units(bytes), Expensive(TRUE)	Specifies the size (bytes) of the blocks that form this StorageExtent. If variable block size, then the maximum block size in bytes should be specified. If the block size is unknown or if a block concept is not valid, enter a 1.
Capacity	Uint64	Units(Bytes)	Specifies the total capacity of the BackendVolume
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	Identifies the object with a short (one-line string) textual description

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
CreationClassName	String	MaxLen(256)	Indicates the name of the class or subclass used to create an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
ConsumableBlocks	Uint64	Expensive(TRUE)	The maximum number of blocks, of size BlockSize, that are available for consumption when layering StorageExtents using the BasedOn association.
ControllerName	String		The name of the volume's backend controller.
DataOrganization	Uint16		Defines the type of data organization techniques: **Code** **Semantics** **0** Other **1** Unknown **2** Fixed Block **3** Variable Block **4** Count Key Data
DataRedundancy	Uint16	ModelCorrespondence (CIM_StorageSetting. DataRedundancyGoal CIM_StorageSetting. DataRedundancyMax CIM_StorageSetting. DataRedundancyMin)	Specifies the number of complete copies of data maintained
DeltaReservation	Uint8	MinValue(0), MaxValue(100), Units(Percentage), ModelCorrespondence (CIM_StorageSetting. DeltaReservationGoal CIM_StorageSetting. DeltaReservationMax CIM_StorageSetting. DeltaReservationMin)	Indicates the current value for the Delta reservation
Description	String		Provides a textual description of the object
DeviceID	String	MaxLen(64)	The ID of the BackendVolume. A numerical value which is unique inside the BackendVolume class only.

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	The BackendVolume's user-friendly name
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicating whether the element is currently shutting down or is in an enabled or disabled state. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
ErrorCleared	Boolean	Deprecated(CIM_Managed SystemElement. OperationalStatus)	Unsupported Property
ErrorDescription	String	Deprecated(CIM_Device ErrorData.ErrorDescription)	Unsupported Property
ErrorMethodology	String		Unsupported Property

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
ExtentStatus	Uint16[]		Records status information in addition to what is captured in the Availability and StatusInfo properties, inherited from ManagedSystem Element. **Code** **Semantics** **0** Other **1** Unknown **2** None/Not Applicable **3** Broken **4** Data Lost **5** Dynamic Reconfig **6** Exposed **7** Fractionally Exposed **8** Partially Exposed **9** Protection Disabled **10** Readying **11** Rebuild **12** Recalculate **13** Spare in Use **14** Verify In Progress **15..32767** DMTF Reserved **32768..65535** Vendor Reserved
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice.Other IdentifyingInfo)	An array of free-form strings providing explanations and details behind the entries in the OtherIdentifyingInfo array. **Note:**each entry of this array is related to the entry in OtherIdentifyingInfo which is located in the same index.
InstallDate	Date-time		Unsupported Property

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
IsBasedOn Underlying Redundancy	Boolean		If set to true, indicates that the underlying StorageExtent participates in a StorageRedundancy Group
LastErrorCode	Uint32	Deprecated(CIM_Device ErrorData.LastErrorCode)	Unsupported Property
MaxPathCount	Uint32	Counter(TRUE), Expensive(TRUE)	Specifies the maximum fibre-channel path count to the BackendVolume
MaxQuiesceTime	Uint64	Deprecated(No value), Units(MilliSeconds)	Unsupported Property
Mode	Uint32	ValueMap, Values	Specifies the mode of the BackendVolume: **Code** **Semantics** **0** Unmanaged **1** Router restricted **2** Managed **3** Image **4** for future use **5** for future use **6** Router config **7** Remote copy **8** for future use
Name	String	MaxLen(1024)	Defines the unique label by which the object is known
NativeStatus	Uint16		The back-end volume's native operational status. **Code** **Semantics** **0** Offline **1** Online **2** Degraded **3** Excluded
NoSingle PointOfFailure	Boolean	ModelCorrespondence (CIM_StorageSetting. NoSinglePointOfFailure)	Indicates whether the no-single-point-of-failure feature exists
NumberOfBlocks	Uint64	Expensive(TRUE)	Specifies the total number of logically contiguous blocks that form the Extent. The total size of the Extent can be calculated by multiplying BlockSize by NumberOfBlocks. If the BlockSize is 1, this property is the total size of the Extent.

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The status of the volume. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Describes the element's enabled or disabled state when EnabledStatus is set to 1; must be null when EnabledStatus is set to a value other than 1
OtherIdentifyingInfo	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice. IdentifyingDescriptions), Expensive(TRUE)	OtherIdentifyingInfo captures additional data, beyond DeviceID information, that could be used to identify a LogicalDevice. One example would be to hold the OperatingSystem's user-friendly name for the Device in this property.
Package Redundancy	Uint16	ModelCorrespondence (CIM_StorageSetting. PackageRedundancyGoal CIM_StorageSetting.Package RedundancyMax CIM_StorageSetting. PackageRedundancyMin)	The number of disk spindles that can fail without data loss

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
PathCount	Uint32	Counter(TRUE), Expensive(TRUE)	Specifies the current counts of fibre-channel paths to the BackendVolume
PoolID	String		Defines the identifier of the associated storage pool
PoolName	String		Defines the name of the associated storage pool
PowerManagement Capabilities	Uint16[]	Deprecated (CIM_PowerManagement Capabilities.PowerCapabilities)	An enumerated array describing the power management capabilities of the device. The use of this property has been deprecated. Instead, the PowerCapabilites property in an associated PowerManagement Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
PowerManagement Supported	Boolean	Deprecated (CIM_PowerManagement Capabilities)	Indicates that the Device can be power-managed. The use of this property has been deprecated. Instead, the existence of an associated PowerManagement Capabilities class (associated using the ElementCapabilities relationhip) indicates that power management is supported.

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
PowerOnHours	Uint64	Deprecated (CIM_PoweredStatistical Data.PowerOnHours), Units(Hours), Counter(TRUE)	Unsupported Property
Primordial	Boolean		If true, ″Primordial″ indicates that the containing system does not have the ability to create or delete this operational element.
Purpose	String		A free form string describing the media and/or its use.
QuorumIndex	Uint8	Expensive(TRUE)	Specifies the quorum index of the BackendVolume; valid indexes are 0, 1, 2 while an index of 3 indicates that this volume is not used as a quorum disk
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. This property is provided to compare Requested and current Enabledstatus. Note that SAN Volume Controller does not evaluate this attribute and therefore no action is taken when it is changed. **Code Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 6. IBMTSSVC_BackendVolume properties (continued)

Property	Type	Qualifier	Description
SequentialAccess	Boolean		If set to true, indicates that the storage is sequentially accessed by a MediaAccessDevice. A TapePartition is an example of a sequentially accessed StorageExtent. StorageVolumes, DiskPartitions and LogicalDisks represent randomly accessed Extents.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
StatusInfo	Uint16	Deprecated (CIM_EnabledLogical Element.EnabledState)	Deprecated property **Code** **Semantics** **1** Other **2** Unknown **3** Enabled **4** Disabled **5** Not Applicable
SystemCreation ClassName	String	Propagated, Key, MaxLen(256)	The scoping system's CreationClassName
SystemName	String	Propagated, Key, MaxLen(256)	The scoping cluster's IP address
TimeOfLast StateChange	Date-time		Unsupported Property
TotalPowerOnHours	Uint64	Deprecated (CIM_PoweredStatistical Data.TotalPowerOnHours), Units(Hours), Counter(TRUE)	Unsupported Property

IBMTSSVC_CandidateCluster

The IBMTSSVC_CandidateCluster class extends the CIM_LogicalElement class.

Properties

The IBMTSSVC_CandidateCluster class represents other IBMTSSVC_Cluster instances that are visible in the fibre-channel SAN and potential candidates for creating a synchronous copy partnership. The IBMTSSVC_CandidateCluster class extends the CIM_LogicalElement class and has the properties shown in Table 7 on page 60.

Table 7. IBMTSSVC_CandidateCluster properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
Description	String		Provides a textual description of the object
ElementName	String		Specifies the name of the cluster
InstallDate	Date-time		Unsupported Property
IsConfigured	Boolean		Partnership configuration state
Name	String	MaxLen(1024)	The label by which the object is known. Format: cluster_ip: candidate_id
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	This property is deprecated in lieu of OperationalStatus
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″)

IBMTSSVC_CandidateNode

The IBMTSSVC_CandidateNode class extends the CIM_ComputerSystem class.

Properties

The IBMTSSVC_CandidateNode class represents a single SAN Volume Controller node in a fibre-channel SAN that is not a member of a cluster, but is available to be a member. The IBMTSSVC_CandidateNode class extends the CIM_ComputerSystem class and has the properties shown in Table 8.

Table 8. IBMTSSVC_CandidateNode properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	The Caption property is a short textual description (one-line string) of the object.

Table 8. IBMTSSVC_CandidateNode properties (continued)

Property	Type	Qualifier	Description
CreationClassName	String	MaxLen(256)	CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Dedicated	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherDedicatedDescriptions)	Enumeration indicating whether the ComputerSystem is a special-purpose System (ie, dedicated to a particular use), versus being general purpose. **Code** **Semantics** **0** Not Dedicated **1** Unknown **2** Other **3** Storage **4** Router **5** Switch **6** Layer 3 Switch **7** Central Office Switch **8** Hub **9** Access Server **10** Firewall **11** Print **12** I/O **13** Web Caching **14** Management **15** Block Server **16** File Server **17** Mobile User Device **18** Repeater **19** Bridge/ Extender **20** Gateway
Description	String		The Description property provides a textual description of the object.
ElementName	String		A user-friendly name for the object.

Table 8. IBMTSSVC_CandidateNode properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	EnabledState is an integer enumeration indicating whether the element is currently shutting down (value = 4), or in an enabled (value = 2) or disabled (value = 3) state. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 8. IBMTSSVC_CandidateNode properties (continued)

Property	Type	Qualifier	Description
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherIdentifyingInfo)	An array of free-form Strings providing explanations and details behind the entries in the OtherIdentifying Info array. **Note:** each entry of this array is related to the entry in OtherIdentifyingInfo that is located at the same index.
InstallDate	Date-time		Unsupported property.
Name	String	MaxLen(256)	The label by which the object is known. Format: cluster_ip:node_id.
NameFormat	String	MaxLen(64)	The NameFormat property identifies how the ComputerSystem Name is generated, using a heuristic. The heuristic is outlined, in detail, in the CIM V2 System Model specification.

Table 8. IBMTSSVC_CandidateNode properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status(es) of the element. Various health and operational statuses are defined. Many of the enumeration's values are self-explanatory. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherDedicated Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. Dedicated)	A string describing how or why the system is dedicated when the Dedicated array includes the value 2, ″Other″.
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.

Table 8. IBMTSSVC_CandidateNode properties (continued)

Property	Type	Qualifier	Description
OtherIdentifying Info	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. IdentifyingDescriptions)	OtherIdentifyingInfo captures additional data, beyond System Name information, that could be used to identify a ComputerSystem. One example would be to hold the Fibre Channel World-Wide Name (WWN) of a node. Note that if only the Fibre Channel name is available and is unique (able to be used as the System key), then this property would be NULL and the WWN would become the System key, its data placed in the Name property.
PowerManagement Capabilities	Uint16[]	Deprecated (CIM_PowerManagement Capabilities.PowerCapabilities)	An enumerated array describing the power management capabilities of the ComputerSystem. The use of this property has been deprecated. Instead, the PowerCapabilites property in an associated PowerManagement Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property.
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property.

Table 8. IBMTSSVC_CandidateNode properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	RequestedState is an integer enumeration indicating whether the element should be shut down (value = 4), enabled (2), disabled (3), taken offline (6) or tested (7) at the next opportunity. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
ResetCapability	Uint16		If enabled, the ComputerSystem can be reset through hardware (e.g. the power and reset buttons). If disabled, hardware reset is not allowed. **Code** **Semantics** **1** Other **2** Unknown **3** Disabled **4** Enabled **5** Not Implemented
Roles	String[]	Write(TRUE)	Unsupported property.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	This property is deprecated in lieu of OperationalStatus.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A String describing the status - used when the OperationalStatus property is set to 1 (″Other″).
TimeOfLastState Change	Date-time		Unsupported property.

IBMTSSVC_CandidateStorageHardwareID

The IBMTSSVC_CandidateStorageHardwareID extends the CIM_LogicalElement class.

Properties

The IBMTSSVC_CandidateStorageHardwareID extends the CIM_LogicalElement class and has the properties shown in Table 9.

Table 9. IBMTSSVC_CandidateStorageHardwareID properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Unsupported property.
Description	String		A textual description of the object.
ElementName	String		Unsupported property.
InstallDate	Date-time		A datetime value indicating when the object was installed. A lack of a value does not indicate that the object is not installed.
Name	String	MaxLen(1024)	The label by which the object is known. When subclassed, the Name property can be overridden to be a Key property.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode **..** DMTF Reserved **0x8000..** Vendor Reserved

Table 9. IBMTSSVC_CandidateStorageHardwareID properties (continued)

Property	Type	Qualifier	Description
Status	String	Deprecated (CIM_ManagedSystem Element.OperationalStatus), MaxLen(10)	A string indicating the current status of the object.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	Strings describing the various OperationalStatus array values.
StorageID	String	ModelCorrespondence (CIM_StorageHardware ID.IDType)	The unique ID of the candidate port
SystemName	String		The IP address of the cluster the candidate to which hwid belongs.

IBMTSSVC_CandidateVolume

The IBMTSSVC_CandidateVolume class represents a potential volume for a synchronous copy relationship.

Properties

The IBMTSSVC_CandidateVolume class extends the CIM_LogicalElement class and has the properties shown in Table 10.

Table 10. IBMTSSVC_CandidateVolume properties

Property	Type	Qualifier	Description
AuxClusterID	String		The ID of this volume's cluster.
Caption	String	MaxLen(64)	A short textual description (one-line string) of the object.
Description	String		A textual description of the object.
ElementName	String		A user-friendly name for the object.
InstallDate	Date-time		Unsupported property.
Name	String	MaxLen(1024)	Defines the label by which the object is known.

Table 10. IBMTSSVC_CandidateVolume properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. Various health and operational statuses are defined. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
SourceVolumeID	String		The ID of the potential master StorageVolume.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	This property is deprecated in lieu of OperationalStatus.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
SystemName	String		The IP address of the scoping cluster.

IBMTSSVC_Chassis

The IBMTSSVC_Chassis class represents the physical frame that contains other elements and provides definable functionality for a product.

Properties

The IBMTSSVC_Chassis class extends the CIM_Chassis class and has the properties shown in Table 11 on page 70.

Table 11. IBMTSSVC_Chassis properties

Property	Type	Qualifier	Description
AudibleAlarm	Boolean		Indicates whether the Frame is equipped with an audible alarm.
BreachDescription	String	ModelCorrespondence (CIM_PhysicalFrame. SecurityBreach)	Unsupported property.
CableManagement Strategy	String		Unsupported property.
CanBeFRUed	Boolean		Unsupported property.
Caption	String	MaxLen(64)	The Caption property is a short textual description (one-line String) of the object.

Table 11. IBMTSSVC_Chassis properties (continued)

Property	Type	Qualifier	Description
ChassisPackage Type	Uint16	Experimental(TRUE), ModelCorrespondence (CIM_Chassis.Chassis TypeDescription)	ChassisPackage Type indicates the physical form factor for the type of Chassis. **Code** **Semantics** **0** Unknown **1** Other **2** SMBIOS Reserved **3** Desktop **4** Low Profile Desktop **5** Pizza Box **6** Mini Tower **7** Tower **8** Portable **9** LapTop **10** Notebook **11** Hand Held **12** Docking Station **13** All in One **14** Sub Notebook **15** Space-Saving **16** Lunch Box **17** Main System Chassis **18** Expansion Chassis **19** SubChassis **20** Bus Expansion Chassis **21** Peripheral Chassis **22** Storage Chassis **23** SMBIOS Reseved **24** Sealed-Case PC **25** SMBIOS Reserved **..** DMTF Reserved **0x8000..0xFFFF** Vendor Reserved
ChassisType Description	String	Experimental(TRUE), ModelCorrespondence (CIM_Chassis.Chassis PackageType)	A string providing more information on the ChassisPackageType.

Table 11. IBMTSSVC_Chassis properties (continued)

Property	Type	Qualifier	Description
ChassisTypes	Uint16[]	Deprecated (CIM_Chassis.Chassis PackageType CIM_Chassis.Multiple SystemSupport), ArrayType(Indexed), ModelCorrespondence (CIM_Chassis.Type Descriptions)	An enumerated, integer-valued array indicating the type of Chassis. **Code** **Semantics** **1** Other **2** Unknown **3** Desktop **4** Low Profile Desktop **5** Pizza Box **6** Mini Tower **7** Tower **8** Portable **9** LapTop **10** Notebook **11** Hand Held **12** Docking Station **13** All in One **14** Sub-Notebook **15** Space-Saving **16** Lunch Box **17** Main System Chassis **18** Expansion Chassis **19** SubChassis **20** Bus Expansion Chassis **21** Peripheral Chassis **22** Storage Chassis **23** Rack Mount Chassis **24** Sealed-Case PC **25** Multi-system Chassis
CreationClassName	String	MaxLen(256)	Indicates the name of the class or subclass used in the creation of an instance. When used with the other key properties of this class, allows all instances of this class and its subclasses to be uniquely identified.
CurrentRequired OrProduced	Uint16	Units(Amps at 120 Volts)	Unsupported property.
Depth	Real32	Units(Inches)	The depth of the PhysicalPackage in inches.
Description	String		Provides a textual description of the object.

Table 11. IBMTSSVC_Chassis properties (continued)

Property	Type	Qualifier	Description
ElementName	String		The chassis' user-friendly name.
HeatGeneration	Uint16	Units(BTU per Hour)	Unsupported property.
Height	Real32	Units(Inches)	The height of the PhysicalPackage in inches.
HotSwappable	Boolean	Deprecated(PhysicalPackage. RemovalConditions)	A PhysicalPackage is HotSwappable if it is possible to replace the Element with a physically different but equivalent one while the containing Package has power applied to it (ie, is 'on').
InstallDate	Date-time		Unsupported property.
IsLocked	Boolean		Unsupported property.
LockPresent	Boolean		Indicates whether the Frame is protected with a lock.
ManufactureDate	Date-time		Unsupported property.
Manufacturer	String	MaxLen(256)	The name of the organization responsible for producing the PhysicalElement. This may be the entity from whom the Element is purchased, but this is not necessarily true. The latter information is contained in the Vendor property of CIM_Product.
Model	String	MaxLen(256)	The name by which the PhysicalElement is generally known.
MultipleSystem Support	Uint16	Experimental(TRUE)	MultipleSystem Support indicates whether or not this chassis supports multiple systems, for example, server blades. **Code** **Semantics** **0** Unknown **1** True **2** False

Table 11. IBMTSSVC_Chassis properties (continued)

Property	Type	Qualifier	Description
Name	String	MaxLen(1024)	Defines the label by which the object is known. When subclassed, the Name property can be overridden to be a Key property.
NumberOf PowerCords	Uint16		Integer indicating the number of power cords which must be connected to the Chassis, for all the components to operate.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. Various health and operational statuses are defined. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherIdentifying Info[]	String	Write(TRUE)	OtherIdentifyingInfo captures additional data, beyond that of Tag information, that could be used to identify a PhysicalElement.
PartNumber	String	MaxLen(256)	The part number assigned by the organization responsible for producing or manufacturing the PhysicalElement.

Table 11. IBMTSSVC_Chassis properties (continued)

Property	Type	Qualifier	Description
PoweredOn	Boolean		Unsupported property.
RackMountable	Uint16	Experimental(TRUE)	RackMountable indicates whether or not the chassis is Rack Mountable **Code** **Semantics** **0** Unknown **1** True **2** False
Removable	Boolean	Deprecated(PhysicalPackage. RemovalConditions)	A PhysicalPackage is Removable if it is designed to be taken in and out of the physical container in which it is normally found, without impairing the function of the overall packaging.
RemovalConditions	Uint16		Is used to describe the conditions under which a PhysicalPackage can be removed. **Code** **Semantics** **0** Unknown **2** Not Applicable **3** Removable when off **4** Removable when on or off
Replaceable	Boolean	Deprecated(No Value)	A PhysicalPackage is Replaceable if it is possible to replace (FRU or upgrade) the Element with a physically different one.
SecurityBreach	Uint16	ModelCorrespondence (CIM_PhysicalFrame. BreachDescription)	An enumerated, integer-valued property: **Code** **Semantics** **1** Other **2** Unknown **3** No Breach **4** Breach Attempted **5** Breach Successful
SerialNumber	String	MaxLen(256)	A manufacturer-allocated number used to identify the PhysicalElement.
ServiceDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_PhysicalFrame. ServicePhilosophy)	Unsupported property.

Table 11. IBMTSSVC_Chassis properties (continued)

Property	Type	Qualifier	Description
ServicePhilosophy	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_PhysicalFrame. ServiceDescriptions)	Unsupported property. **Code** **Semantics** **0** Unknown **1** Other **2** Service From Top **3** Service From Front **4** Service From Back **5** Service From Side **6** Sliding Trays **7** Removable Sides **8** Moveable
SKU	String	MaxLen(64)	Unsupported property.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystemElement. OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for information.
StatusDescriptions[]	String	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
Tag	String	MaxLen(256)	An arbitrary string that uniquely identifies the PhysicalElement and serves as the Element's key.
TypeDescriptions	String[]	Deprecated (CIM_Chassis.Chassis TypeDescription), ArrayType(Indexed), ModelCorrespondence (CIM_Chassis.ChassisTypes)	Unsupported property.
UserTracking	String	Write(TRUE)	Unsupported property.
VendorEquipment Type	String		Unsupported property.
Version	String	MaxLen(64)	A string indicating the version of the PhysicalElement.
VisibleAlarm	Boolean		Indicates that the equipment includes a visible alarm.
Weight	Real32	Units(Pounds)	The weight of the PhysicalPackage in pounds.
Width	Real32	Units(Inches)	The width of the PhysicalPackage in inches.

IBMTSSVC_Cluster

The IBMTSSVC_Cluster class represents a single SAN Volume Controller cluster, with up to four node pairs.

Properties

The IBMTSSVC_Cluster class extends the IBMTSSVC_AbstractCluster class and has the properties shown in Table 12.

Table 12. IBMTSSVC_Cluster properties

Property	Type	Qualifier	Description
AllocatedCapacity	Uint64	Units(Bytes), Expensive(TRUE)	The total capacity of all StorageVolumes in the cluster.
AvailableCapacity	Uint64	Units(Bytes), Expensive(TRUE)	The currently available space in the cluster. This is approximate. BackendStorage Capacity-Allocated Capacity.
Backendstorage Capacity	Uint64	Units(Bytes), Expensive(TRUE)	The total capacity of all backend storage connected to the cluster.
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator), ReadRole(None)	The Caption property is a short textual description (one-line string) of the object.
ClusterState	Uint16		Indicates the state of the Cluster. **Code** **Semantics** **0** Unknown **1** Other **2** On-line **3** Off-line **4** Degraded **5** Unavailable
CodeLevel	String	Expensive(TRUE)	The code level of the cluster.
ConsoleIP	String	Expensive(TRUE), Write(TRUE), WriteRole(Administrator), ReadRole(None)	The IP address of the management console.
ConsolePort	String	Expensive(TRUE), Write(TRUE), WriteRole(Administrator), ReadRole(None)	The port address of the management console.
CreationClassName	String	MaxLen(256), ReadRole(None)	Indicates the name of the class or subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.

Table 12. IBMTSSVC_Cluster properties (continued)

Property	Type	Qualifier	Description
Dedicated	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherDedicatedDescriptions)	Enumeration indicating whether the ComputerSystem is a special-purpose System (dedicated to a particular use), versus being general purpose. SAN Volume Controller is a dedicated storage device and will return {3,15} (″Storage,″ ″Block Server″) **Code** **Semantics** **0** Not Dedicated **1** Unknown **2** Other **3** Storage **4** Router **5** Switch **6** Layer 3 Switch **7** Central Office Switch **8** Hub **9** Access Server **10** Firewall **11** Print **12** I/O **13** Web Caching **14** Management **15** Block Server **16** File Server **17** Mobile User Device **18** Repeater **19** Bridge/ Extender **20** Gateway
Description	String		Provides a textual description of the object.
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator), ReadRole(None)	The cluster's user-friendly name
EmailSetting	String	Expensive(TRUE)	The email setting.

Table 12. IBMTSSVC_Cluster properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE),	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	An integer enumeration indicator. In various situations, an element that is being tested is neither enabled or disabled — this is addressed by the value ″In Test″ (7). If this property does not apply to an instance of EnabledLogical Element, the value 5 is used. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
FcPortSpeed	Uint64	Units(GigaBit per second), Expensive(TRUE)	The transmission speed of the attached Fibre-Channel.

Table 12. IBMTSSVC_Cluster properties (continued)

Property	Type	Qualifier	Description
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherIdentifyingInfo)	An array of free-form strings providing explanations and details behind the entries in the OtherIdentifying Info array. Note that each entry of this array is related to the entry in OtherIdentifying Info that is located at the same index.
InstallDate	Date-time		Unsupported property.
Interconnect	String		Unsupported Property.
InterconnectAddress	String		Unsupported Property.
Locale	String	Expensive(TRUE)	The current locale setting of the cluster.
MaxNumberOf Nodes	Uint32		Indicates the maximum number of nodes that may participate in the Cluster. If unlimited, enter 0.
Name	String	MaxLen(256), ReadRole(None)	The label by which the object is known. Format: cluster_ip:object_id
NameFormat	String	MaxLen(64)	Identifies how the ComputerSystem Name is generated. SAN Volume Controller returns the cluster's id as Name, therefore this attribute is set to ″Other.″

Table 12. IBMTSSVC_Cluster properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The cluster's operational status. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherDedicated Descriptions	String[]	ModelCorrespondence (CIM_ComputerSystem. Dedicated), ArrayType(Indexed)	A string describing how or why the system is dedicated when the Dedicated array includes the value 2, ″Other.″
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
OtherIdentifying Info	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. IdentifyingDescriptions), Write(TRUE), WriteRole(Service), Expensive(TRUE)	The cluster's IP Address, Subnet Mask, Default Gateway and Service IP Address.
PoolCapacity	Uint64	Units(Bytes), Expensive(TRUE)	The total capacity of all StoragePools in the cluster.

Table 12. IBMTSSVC_Cluster properties (continued)

Property	Type	Qualifier	Description
PowerManagement Capabilities	Uint16[]	Deprecated (CIM_PowerManagement Capabilities.PowerCapabilities)	An enumerated array describing the power management capabilities of the ComputerSystem. The use of this property has been deprecated. Instead, the Power Capabilites property in an associated PowerManagement Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE), Expensive(TRUE)	The EMail address of the primary contact person for this cluster
PrimaryOwner Name	String	MaxLen(64), Write(TRUE),	Unsupported Property
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration value. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
RequiredMemory	Uint32	Units(MegaBytes), Expensive(TRUE)	The amount of required memory for that cluster.

Table 12. IBMTSSVC_Cluster properties (continued)

Property	Type	Qualifier	Description
ResetCapability	Uint16		If enabled (value = 4), the ComputerSystem can be reset through hardware (the power and reset buttons). If disabled (value = 3), hardware reset is not allowed. **Code** **Semantics** **1** Other **2** Unknown **3** Disabled **4** Enabled **5** Not Implemented
Roles	String[]	Write(TRUE)	Unsupported Property
SNMPCommunity	String	Expensive(TRUE)	The SNMP cummunity.
SNMPServerIP	String	Expensive(TRUE)	The SNMP server IP address.
SNMPSetting	String	Expensive(TRUE)	The SNMP setting of the cluster.
StatisticsFrequency	Uint32	Units(Seconds), Expensive(TRUE)	Indicates the update interval for the cluster statistics
StatisticsStatus	Boolean	Expensive(TRUE)	Is true if statistics collection is active.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	The status of the cluster. Can be either ″OK,″ ″Key Refused,″ ″Invalid Fingerprint,″ ″Cluster Interface Not Available,″ ″CLI Error rc,″ ″Connection Lost,″ or ″No Contact.″
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
TimeOfLast StateChange	Date-time		Unsupported property.
TimeZone	String	Expensive(TRUE)	The timezone setting of the cluster.

Table 12. IBMTSSVC_Cluster properties (continued)

Property	Type	Qualifier	Description
Types	Uint16[]		The cluster types. This specifies whether the cluster is for failover (value=2), performance (3), etc. The values which can be specified are not mutually exclusive. Thus, Types is an array. **Code** **Semantics** **0** Unknown **1** Other **2** Failover **3** Performance **4** Distributed OS **5** Node Grouping **6** SysPlex

IBMTSSVC_Controller

The IBMTSSVC_Controller class extends the CIM_SCSIController class.

Properties

The IBMTSSVC_Controller class represents the logical SAN Volume Controller used for modeling the authorization path from host ports to storage volumes and has the properties shown in Table 13.

Table 13. IBMTSSVC_Controller properties

Property	Type	Qualifier	Description
AccessGranted	Boolean	ModelCorrespondence (CIM_Controller. AuthorizationView)	This property provides a quick interface for finding Devices with no Authorization Subject association to an AccessControl Information instance; either directly, or via a Controller. True indicates that the Device has granted access to some consumer. False indicates that no access has been granted.

Table 13. IBMTSSVC_Controller properties (continued)

Property	Type	Qualifier	Description
Additional Availability	Uint16[]	Deprecated(CIM_Associated PowerManagementService. PowerState CIM_ManagedSystem Element.OperationalStatus CIM_EnabledLogical Element.EnabledStatus), ModelCorrespondence (CIM_LogicalDevice. Availability)	Additional availability and status of the Device, beyond that specified in the Availability property. **Code** **Semantics** **1** Other **2** Unknown **3** Running/ Full Power **4** Warning **5** In Test **6** Not Applicable **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced

Table 13. IBMTSSVC_Controller properties (continued)

Property	Type	Qualifier	Description
Availability	Uint16	Deprecated(CIM_Associated PowerManagementService. PowerState CIM_ManagedSystem Element.OperationalStatus CIM_EnabledLogical Element.EnabledStatus), ModelCorrespondence (CIM_LogicalDevice. AdditionalAvailability)	The primary availability and status of the Device. **Code** **Semantics** **1** Other **2** Unknown **3** Running/ Full Power **4** Warning **5** In Test **6** Not Applicable **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced
Caption	String	MaxLen(64)	Unsupported property.
ConnectionRole	Uint16[]		A protocol controller can have one or more of several roles in a connection:\n. In certain applications, a controller can have both functions (both providing and consuming the connection), hence the array).\nSpecific **Examples:** A HBA which would be a 'client,' a front end controller on a RAID array which would be a 'server' and a SCSI Extended Copy controller which would be both. **Code** **Semantics** **0** Unknown **2** Server **3** Client
Controlled	Boolean		Unsupported property.

Table 13. IBMTSSVC_Controller properties (continued)

Property	Type	Qualifier	Description
CreationClassName	String	MaxLen(256)	CreationClassName indicates the name of the class or subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		The Description property provides a textual description of the object.
DeviceID	String	MaxLen(64)	An address or other identifying information to uniquely name the LogicalDevice.
ElementName	String		Unsupported property.
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 13. IBMTSSVC_Controller properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration. In various situations, an element that is being tested is neither enabled or disabled - this is addressed by the value ″In Test″ (7). If this property does not apply to an instance of EnabledLogical Element, the value 5 (″Not Applicable″) is used. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
ErrorCleared	Boolean	Deprecated(CIM_Managed SystemElement. OperationalStatus)	Unsupported property.
ErrorDescription	String	Deprecated(CIM_Device ErrorData.ErrorDescription)	Unsupported property.
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice. OtherIdentifyingInfo)	An array of free-form strings providing explanations and details behind the entries in the OtherIdentifyingInfo array. **Note:** each entry of this array is related to the entry in OtherIdentifying Info that is located at the same index.
InstallDate	Date-time		Unsupported property.
LastErrorCode	Uint32	Deprecated(CIM_Device ErrorData.LastErrorCode)	Unsupported property.
MaxQuiesce Time	Uint64	Deprecated(No value), Units(MilliSeconds)	Unsupported property.

Table 13. IBMTSSVC_Controller properties (continued)

Property	Type	Qualifier	Description
MaxUnits Controlled	Uint32		Maximum number of Units that can be controlled by or accessed through this protocol controller.
Name	String	MaxLen(1024)	The label by which the object is known. Format: RedundancyGroup_id: Host_id.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. This property always reports ″Unknown.″ **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	String describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
OtherIdentifying Info	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice. IdentifyingDescriptions)	OtherIdentifying Info captures additional data, beyond DeviceID information, that could be used to identify a LogicalDevice.

Table 13. IBMTSSVC_Controller properties (continued)

Property	Type	Qualifier	Description
Power Management Capabilities	Uint16[]	Deprecated(CIM_Power ManagementCapabilities. PowerCapabilities)	The use of this property has been deprecated. PowerCapabilites property in an associated Power Management Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
Power Management Supported	Boolean	Deprecated(CIM_Power ManagementCapabilities)	Boolean indicating that the Device can be power managed.
PowerOnHours	Uint64	Deprecated(CIM_Powered StatisticalData.PowerOn Hours), Units(Hours), Counter(TRUE)	Unsupported property.
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Status	String	MaxLen(10), Deprecated(CIM_Managed SystemElement.Operational Status)	Deprecated property; set to ″Unknown.″
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status — used when the OperationalStatus property is set to 1 (″Other″)

Table 13. IBMTSSVC_Controller properties (continued)

Property	Type	Qualifier	Description
StatusInfo	Uint16	Deprecated(CIM_Enabled LogicalElement.EnabledState)	Indicates whether the LogicalDevice is in an enabled or disabled state. StatusInfo does not apply to the Controller, so the value ″Not Applicable″ is used. **Code** **Semantics** **1** Other **2** Unknown **3** Enabled **4** Disabled **5** Not Applicable
SystemCreation ClassName	String	Propagated(CIM_System. CreationClassName), MaxLen(256)	The scoping System's CreationClassName.
SystemName	String	Propagated(CIM_System.Name), MaxLen(256)	The scoping System's Name.
TimeOfLast StateChange	Date-time		Unsupported property.
TotalPower OnHours	Uint64	Deprecated(CIM_Powered StatisticalData.TotalPower OnHours), Units(Hours), Counter(TRUE)	Unsupported property.

IBMTSSVC_ControllerConfigurationService

The IBMTSSVC_ControllerConfigurationService class extends the CIM_ControllerConfigurationService.

Properties

The IBMTSSVC_ControllerConfigurationService class has the properties shown in Table 14.

Table 14. IBMTSSVC_ControllerConfigurationService properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64),	Unsupported property.
CreationClassName	String	MaxLen(256)	CreationClass Name indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		The Description property provides a textual description of the object.

Table 14. IBMTSSVC_ControllerConfigurationService properties (continued)

Property	Type	Qualifier	Description
ElementName	String		Unsupported property
EnabledDefault	Uint16	Write(TRUE)	Enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled.″ **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator. **Code** Semantics **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(256)	The label by which the object is known.

Table 14. IBMTSSVC_ControllerConfigurationService properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The operational status of the service. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabled State	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	String describing the element's enabled/disabled state when the EnabledStatus property is set to 1 (″Other″). This property MUST be set to NULL when EnabledStatus is any value other than 1.
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property.
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property.

Table 14. IBMTSSVC_ControllerConfigurationService properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. Note that when EnabledStatus is set to 5 (″Not Applicable″), writing this property has no effect. By default, the element's RequestedStatus is set to″No Change.″ **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Started	Boolean		Indicates if this service is started.
StartMode	String	MaxLen(10), Deprecated(CIM_Service. EnabledDefault)	Indicates if this service is started manually or automatic.
Status	String	MaxLen(10), Deprecated(CIM_Managed SystemElement.Operational Status)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
SystemCreation ClassName	String	Propagated(CIM_System. CreationClassName), MaxLen(256)	The scoping system's creation class name.
SystemName	String	Propagated(CIM_System. Name), MaxLen(256)	The scoping system's name.
TimeOfLast StateChange	Date-time		Unsupported property.

IBMTSSVC_ControllerMaskingCapabilities

The IBMTSSVC_ControllerMaskingCapabilities class extends the CIM_ProtocolControllerMaskingCapabilities class.

Properties

The IBMTSSVC_ControllerMaskingCapabilities class has the properties shown in Table 15.

Table 15. IBMTSSVC_ControllerMaskingCapabilities properties

Property	Type	Qualifier	Description
AttachDevice Supported	Boolean		Set to true if this storage system supports the AttachDevice method.
Caption	String	MaxLen(64)	Short textual description (one-line string) of the object.
ClientSelectable DeviceNumbers	Boolean		Set to true if this storage system allows the client to specify the DeviceNumber parameter when calling Controller Configuration Service.AttachDevice() and Controller Configuration Service.AttachDevice(). Set to false if the implementation does not allow unit numbers to vary for a Protocol Controller.
Description	String		Provides a textual description of the object.
ElementName	String	Required(TRUE)	The user-friendly name for this instance of Capabilities.
InstanceID	String		Opaquely and uniquely identifies an instance of this class.
OneHardware IDPerView	Boolean		Set to true if this storage system limits configurations to a single subject hardware ID per view. Otherwise, multiple hardware ID types can be used. The default is FALSE, in that multiple ID types may be used in a single view.
OtherValid HardwareID Types	String[]	ArrayType(Indexed)	An array of strings describing types for valid Storage HardwareID. IDType. Used when the ValidHardware IdTypes includes 1 (″Other″)

Table 15. IBMTSSVC_ControllerMaskingCapabilities properties (continued)

Property	Type	Qualifier	Description
PortsPerView	Uint16		An integer enumeration indicating the way that ports per view (ProtocolController) are managed by the underlying storage system. **Code** **Semantics** **2** One Port per View **3** Multiple Ports per View **4** All Ports share the same View
PrivilegeDenied Supported	Boolean		Set to true if this storage system allows a client to create a Privilege instance with PrivilegeGranted set to FALSE.
Protocol Controller Requires AuthorizedIdentity	Boolean		If true, this property indicates that at least one Privilege/Identity pair must be specified when CreateProtocol Controller() is called.
Protocol Controller Supports Collections	Boolean		If true, this property indicates that the Identity parameter of CreateProtocol Conntroller WithPorts() MUST contain a reference to a CIM_Collection (or subclass) or to a CIM_Identity (or subclass).
UniqueUnit NumbersPerPort	Boolean		When set to false, different Protocol Contollers attached to a LogicalPort can expose the same unit numbers. If true, then this storage system requires unique unit numbers across all the Protocol Controllers connected to a LogicalPort.

Table 15. IBMTSSVC_ControllerMaskingCapabilities properties (continued)

Property	Type	Qualifier	Description
ValidHardware IdTypes	Uint16[]	ArrayType(Indexed)	A list of the valid values for StrorageHardware ID.IDType. **Code** **Semantics** **1** Other **2** Port WWN **3** Node WWN **4** Host Name

IBMTSSVC_Dumps

The IBMTSSVC_Dumps class extends the CIM_SettingData class.

Properties

The IBMTSSVC_Dumps class has the properties shown in Table 16.

Table 16. IBMTSSVC_Dumps properties

Property	Type	Qualifier	Description
Admin	String[]		An array of admin log file names.
Caption	String	MaxLen(64)	The Caption property is a short textual description (one-line string) of the object.
Configs	String[]		An array of config dump file names.
Description	String		Provides a textual description of the object.
ElementName	String	Required(TRUE)	The user-friendly name for this instance of SettingData.
Elogs	String[]		An array of error log file names.
Feature	String[]		An array of feature log file names.

Table 16. IBMTSSVC_Dumps properties (continued)

Property	Type	Qualifier	Description
InstanceID	String		Opaquely identifies a unique instance of SettingData. The InstanceID must be unique within a namespace. In order to ensure this, the value of InstanceID SHOULD be constructed in the following manner: \n(Vendor ID)(ID) \n (Vendor ID) MUST include a copyrighted, trademarked or otherwise unique name that is owned by the business entity or a registered ID that is assigned to the business entity that is defining the InstanceID.
Iostats	String[]		An array of iostats log file names.
Iotrace	String[]		An array of iotrace log file names.

IBMTSSVC_FCPort

The IBMTSSVC_FCPort class represents a fibre-channel port of a SAN Volume Controller node.

Properties

Generally, all fibre-channel ports of a SAN Volume Controller pair are connected to the same devices. The IBMTSSVC_FCPort class extends the CIM_FCPort class and has the properties shown in Table 17.

Table 17. IBMTSSVC_FCPort properties

Property	Type	Qualifier	Description
ActiveCOS	Uint16[]	ModelCorrespondence (CIM_FCPort.SupportedCOS)	An array of integers indicating the Classes of Service that are active (COS). The Active COS is indicated in ActiveCOS. **Code** **Semantics** **0** Unknown **1** 1 **2** 2 **3** 3 **4** 4 **5** 5 **6** 6 **7** F

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
ActiveFC4Types	Uint16[]	ModelCorrespondence (CIM_FCPort.Supported FC4Types)	Array of integers indicating the Fibre Channel FC-4 protocols currently running. A list of all protocols supported is indicated in the Supported FC4Types property. **Code** **Semantics** **0** Unknown **1** Other **4** ISO/IEC 8802 - 2 LLC **5** IP over FC **8** SCSI - FCP **9** SCSI - GPP **17** IPI - 3 Master **18** IPI - 3 Slave **19** IPI - 3 Peer **21** CP IPI - 3 Master **22** CP IPI - 3 Slave **23** CP IPI - 3 Peer **25** SBCCS Channel **26** SBCCS Control Unit **27** FC-SB-2 Channel **28** FC-SB-2 Control Unit **32** Fibre Channel Services (FC-GS, FC-GS-2, FC-GS-3) **34** FC-SW **36** FC - SNMP **64** HIPPI - FP **80** BBL Control **81** BBL FDDI Encapsulated LAN PDU **82** BBL 802.3 Encapsulated LAN PDU **88** FC - VI **96** FC - AV **255** Vendor Unique
ActiveMaximum TransmissionUnit	Uint64	Units(Bytes),	Unsupported property.

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
Additional Availability	Uint16[]	Deprecated (CIM_AssociatedPower ManagementService. PowerState CIM_ManagedSystem Element.OperationalStatus CIM_EnabledLogical Element.EnabledState), ModelCorrespondence (CIM_LogicalDevice. Availability)	Additional availability and status of the Device, beyond that specified in the Availability property. Will be always reported as ″Unknown″ by the SAN Volume Controller.
AutoSense	Boolean		Indicates whether the NetworkPort is capable of automatically determining the speed or other communications characteristics of the attached network media.
Availability	Uint16	Deprecated (CIM_AssociatedPower ManagementService. PowerState CIM_ManagedSystem Element.OperationalStatus CIM_EnabledLogical Element.EnabledState) ModelCorrespondence (CIM_LogicalDevice. AdditionalAvailability)	The primary availability and status of the Device. Will be always reported as ″Unknown.″
Caption	String	MaxLen(64)	Unsupported property.
CreationClassName	String	MaxLen(256)	CreationClass Name indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		Provides a textual description of the object.
DeviceID	String	MaxLen(64), Expensive(TRUE)	An address or other identifying information to uniquely name the LogicalDevice.
ElementName	String		Unsupported property.

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	Enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled.″ **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
ErrorCleared	Boolean	Deprecated (CIM_ManagedSystem Element.OperationalStatus)	Unsupported property.
ErrorDescription	String	Deprecated (CIM_DeviceErrorData. ErrorDescription)	Unsupported property.
FullDuplex	Boolean		Boolean indicating that the port is operating in full duplex mode.

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice. OtherIdentifyingInfo)	An array of free-form strings providing explanations and details behind the entries in the OtherIdentifying Info array. Note, each entry of this array is related to the entry in OtherIdentifyingInfo that is located at the same index.
InstallDate	Date-time		Unsupported property.
LastErrorCode	Uint32	Deprecated (CIM_DeviceErrorData. LastErrorCode)	Unsupported property.
LinkTechnology	Uint16	ModelCorrespondence (CIM_NetworkPort.Other LinkTechnology)	An enumeration of the types of links. When set to 1 (″Other″), the related property OtherLink Technology contains a string description of the link's type. **Code** **Semantics** **0** Unknown **1** Other **2** Ethernet **3** IB **4** FC **5** FDDI **6** ATM **7** Token Ring **8** Frame Relay **9** Infrared **10** BlueTooth **11** Wireless LAN
MaxQuiesceTime	Uint64	Deprecated(No value), Units(MilliSeconds)	Unsupported property.
MaxSpeed	Uint64	Units(Bits per Second)	The max speed of the Port in Bits per Second.
Name	String	MaxLen(1024)	Unsupported property.
NetworkAddresses	String[]	MaxLen(64)	Unsupported property.

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions), Expensive(TRUE)	Indicates the current status of the port. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabled State	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
OtherIdentifying Info	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice. IdentifyingDescriptions)	Captures additional data, beyond DeviceID information, that could be used to identify a LogicalDevice.
OtherLink Technology	String	ModelCorrespondence (CIM_NetworkPort. LinkTechnology)	A string value describing LinkTechnology when it is set to ″Other.″
OtherNetwork PortType	String	ModelCorrespondence (CIM_NetworkPort. PortType)	Describes the type of module, when PortType is set to ″Other.″
Permanent Address	String	MaxLen(64), Expensive(TRUE)	Defines the network address hardcoded into a port. This 'hardcoded' address may be changed through firmware upgrade or software configuration.

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
PortNumber	Uint16	Expensive(TRUE)	NetworkPorts are often numbered relative to either a logical modules or a network element.
PortType	Uint16	ModelCorrespondence (CIM_NetworkPort. OtherNetworkPortType)	The specific mode currently enabled for the Port. **Code** **Semantics** **0** Unknown **1** Other **10** N **11** NL **12** F/NL **13** Nx **14** E **15** F **16** FL **17** B **18** G **16000..65535** Vendor Reserved
Power Management Capabilities	Uint16[]	Deprecated (CIM_PowerManagement Capabilities.Power Capabilities)	Describes the power management capabilities of the Device. The use of this property has been deprecated. Instead, the PowerCapabilites property in an associated Power Management Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
Power Management Supported	Boolean	Deprecated (CIM_PowerManagement Capabilities)	Boolean indicating that the Device can be power managed. The use of this property has been deprecated. Instead, the existence of an associated Power Management Capabilities class (associated using the Element Capabilities relationship) indicates that power management is supported.
PowerOnHours	Uint64	Deprecated (CIM_PoweredStatistical Data.PowerOnHours), Units(Hours), Counter(TRUE)	Unsupported property.
RequestedSpeed	Uint64	Write(TRUE), Units(Bits per Second), ModelCorrespondence (CIM_LogicalPort.Speed)	Unsupported property.
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. Note that SAN Volume Controller does not evaluate this attribute and therefore no action is taken when it is changed. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Speed	Uint64	Units(Bits per Second)	An estimate of the current bandwidth in Bits per Second. For ports that vary in bandwidth or for those where no accurate estimation can be made, this property should contain the nominal bandwidth.

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	Describes the status - used when the OperationalStatus property is set to ″Other.″
StatusInfo	Uint16	Deprecated (CIM_EnabledLogical Element.EnabledState)	Indicates whether the LogicalDevice is in an enabled, disabled, or some other state. Since this property is deprecated, the value ″Unknown″ is always reported.
SupportedCOS	Uint16[]		An array of integers indicating the Fibre Channel Classes of Service (COS) that are supported. The active COS are indicated in ActiveCOS. **Code** **Semantics** **0** Unknown **1** 1 **2** 2 **3** 3 **4** 4 **5** 6 **6** F

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
Supported FC4Types	Uint16[]		Array of integers indicating the Fibre Channel FC-4 protocols supported. The protocols that are active and running are indicated in the ActiveFC4 Types property. **Code** **Semantics** **0** Unknown **1** Other **4** ISO/IEC 8802 - 2 LLC **5** IP over FC **8** SCSI - FCP **9** SCSI - GPP **17** IPI - 3 Master **18** IPI - 3 Slave **19** IPI - 3 Peer **21** CP IPI - 3 Master **22** CP IPI - 3 Slave **23** CP IPI - 3 Peer **25** SBCCS Channel **26** SBCCS Control Unit **27** FC-SB-2 Channel **28** FC-SB-2 Control Unit **32** Fibre Channel Services (FC-GS, FC-GS-2, FC-GS-3) **34** FC-SW **36** FC - SNMP **64** HIPPI - FP **80** BBL Control **81** BBL FDDI Encapsulated LAN PDU **82** BBL 802.3 Encapsulated LAN PDU **88** FC - VI **96** FC - AV **255** Vendor Unique
SupportedMaximum TransmissionUnit	Uint64	Units(Bytes)	Unsupported property.
SystemCreation ClassName	String	Propagated (CIM_System.Creation ClassName), MaxLen(256)	The scoping system's CreationClass Name.

Table 17. IBMTSSVC_FCPort properties (continued)

Property	Type	Qualifier	Description
SystemName	String	Propagated (CIM_System.Name), MaxLen(256)	The scoping system's Name.
TimeOfLast StateChange	Date-time		Unsupported property.
TotalPowerOnHours	Uint64	Deprecated (CIM_PoweredStatistical Data.TotalPowerOnHours), Units(Hours), Counter(TRUE)	Unsupported property.
UsageRestriction	Uint16		Indicates if the port is restricted for front- or backe-end use. **Code** **Semantics** **0** Unknown **2** Front-end only **3** Back-end only **4** Not restricted

IBMTSSVC_Features

The IBMTSSVC_Features class specifies the enabled capabilities of the SAN Volume Controller.

Properties

The IBMTSSVC_Features instance corresponds with the associated IBMTSSVC_Cluster instance. The IBMTSSVC_Features class extends the CIM_Capabilities class and has the properties shown in Table 18.

Table 18. IBMTSSVC_Features properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Unsupported Property.
Description	String		Provides a textual description of the object.
ElementName	String	Required(TRUE)	Unsupported Property.
FlashCopy	Boolean	Write(TRUE), WriteRole(Administrator)	Specifies whether the FlashCopy feature is enabled for the cluster.

Table 18. IBMTSSVC_Features properties (continued)

Property	Type	Qualifier	Description
InstanceID	String		Opaquely identifies a unique instance of Capabilities. The InstanceID must be unique within a namespace. To ensure this, the value of InstanceID should be constructed in the following manner: \n(VendorID)(ID) \n.
MaximumCapacity	Uint64	Units	Specifies the maximum capacity of the cluster.
RemoteCopy	Boolean	Write(TRUE), WriteRole(Administrator)	Specifies whether the RemoteCopy feature is enabled for the cluster.

IBMTSSVC_FlashCopySynchronizedSet

The IBMTSSVC_FlashCopySynchronizedSet class aggregates multiple IBMTSSVC_StorageSynchronized instances to ensure consistent copying.

Properties

The IBMTSSVC_FlashCopySynchronizedSet class extends the CIM_SynchronizedSet class and has the properties shown in Table 19.

Table 19. IBMTSSVC_FlashCopySynchronizedSet properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	Identifies the object with a short (one-line string) textual description

Table 19. IBMTSSVC_FlashCopySynchronizedSet properties (continued)

Property	Type	Qualifier	Description
CopyType	Uint16	Experimental(TRUE)	Describes the replication policy of the SynchronizedSet. Values are: \n Async: create and maintain an asynchronous \n copy of the source. \n Sync: create and maintain a synchronized copy \n of the source. \n UnSyncAssoc: create an unsynchronized copy and \n maintain an association to the source. **Code** Semantics **2** Async **3** Sync **4** UnSyncAssoc **..** DMTF Reserved **0x8000..** Vendor Specific
Description	String		Provides a textual description of the object.
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator), Experimental(TRUE)	Specifies the user-friendly name for this instance of SynchronizedSet.
InstanceID	String	Key	Opaquely identifies a unique instance that is scoped (contained) by a System. The InstanceID must be unique within a namespace. In order to ensure this, the value of InstanceID should be constructed in the following manner: \n(Vendor ID)(ID) \n.
Status	Uint32		Indicates the status of the SynchronizedSet. **Code** Semantics **2** Initialized **3** PrepareInProgress **4** Prepared **5** ResyncInProgress **11** Idle **12** Broken **0x1000** Empty **0x8001** Stopped

IBMTSSVC_HardwareIdCollection

The IBMTSSVC_HardwareIdCollection class represents a host port to which volume access can be granted.

Properties

The IBMTSSVC_HardwareIdCollection class extends the CIM_SystemSpecificCollection class and has the properties shown in Table 20.

Table 20. IBMTSSVC_HardwareIdCollection properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	A short textual description (one-line string) of the object.
Description	String		Provides a textual description of the object.
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	A user-friendly name for the object.
InstanceID	String	MaxLen(256)	The label by which the object is known. Format: cluster_ip:object_id.
MaxPathCount	Uint32	Counter(TRUE), Expensive(TRUE)	The maximum number of FC paths to this host.
NumberOfPorts	Uint32	Counter(TRUE)	The number of FC ports registered for this host.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystemElement. StatusDescriptions)	Indicates the current status of the element. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
PathCount	Uint32	Counter(TRUE), Expensive(TRUE)	The current number of FC paths to this host.
PortWWN	String[]	Expensive(TRUE)	The FC ports registered for this host.

Table 20. IBMTSSVC_HardwareIdCollection properties (continued)

Property	Type	Qualifier	Description
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to ″Other.″

IBMTSSVC_IOGroup

The IBMTSSVC_IOGroup class extends the CIM_ComputerSystem class.

Properties

The IBMTSSVC_IOGroup class defines an interface for a set of Volumes. All Nodes and Volumes are associated with exactly one IOGroup. The read and write cache provided by a node is duplicated for redundancy. When I/O is performed to a Volume, the node that processes the I/O will duplicate the data on the Partner node in the IOGroup. This class represents the system aspect of an I/O group, wheras IOGroupSet represents the set aspect.

The IBMTSSVC_IOGroup class extends the CIM_ComputerSystem class and has the properties shown in Table 21.

Table 21. IBMTSSVC_IOGroup properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	A short textual description (one-line string) of the object.
CreationClassName	String	MaxLen(256)	Indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.

Table 21. IBMTSSVC_IOGroup properties (continued)

Property	Type	Qualifier	Description
Dedicated	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherDedicatedDescriptions)	Enumeration indicating whether the ComputerSystem is a special-purpose System (dedicated to a particular use) versus being general purpose. SAN Volume Controller is a dedicated storage device and will return {3,15} (″Storage,″ ″Block Server″). **Code** **Semantics** **0** Not Dedicated **1** Unknown **2** Other **3** Storage **4** Router **5** Switch **6** Layer 3 Switch **7** Central Office Switch **8** Hub **9** Access Server **10** Firewall **11** Print **12** I/O **13** Web Caching **14** Manage-ment **15** Block Server **16** File Server **17** Mobile User Device **18** Repeater **19** Bridge/ Extender **20** Gateway
Description	String		Provides a textual description of the object.
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	The group's user-friendly name.

Table 21. IBMTSSVC_IOGroup properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled.″ **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicating whether the element is currently shutting down or in an enabled or disabled state. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherIdentifyingInfo)	An array of free-form strings providing explanations and details behind the entries in the OtherIdentifyingInfo array.
InstallDate	Date-time		Unsupported property.

Table 21. IBMTSSVC_IOGroup properties (continued)

Property	Type	Qualifier	Description
Name	String	MaxLen(256)	The label by which the object is known. Format: cluster_ip:object_id.
NameFormat	String	MaxLen(64)	Identifies how the ComputerSystem Name is generated. The SAN Volume Controller returns the node's id as Name, therefore this attribute is set to ″Other.″
NumberOfNodes	Uint32	Counter(TRUE)	The number of nodes in the group.
NumberOfVolumes	Uint32	Counter(TRUE)	The number of virtual disk offered by the group.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	No status information available on group level. Look for the individual nodes' status. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherDedicated Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. Dedicated)	A string describing how or why the system is dedicated when the Dedicated array includes the value 2 (″Other″).

Table 21. IBMTSSVC_IOGroup properties (continued)

Property	Type	Qualifier	Description
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
OtherIdentifyingInfo	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. IdentifyingDescriptions)	The node's ″Redundancy Group ID,″ ″RedundancyGroup Name,″ ″Partner Node Name,″ ″Partner Node ID″ and ″WWWN.″
PowerManagement Capabilities	Uint16[]	Deprecated (CIM_PowerManagement Capabilities.PowerCapabilities)	An enumerated array describing the power management capabilities of the ComputerSystem. The use of this property has been deprecated. Instead, the PowerCapabilites property in an associated PowerManagement Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Auto-matically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property.
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property.

Table 21. IBMTSSVC_IOGroup properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicating whether the element should be shut down, enabled, disabled, taken offline, or tested at the next opportunity. This property is provided to compare Requested and current Enabledstatus. Note that the SAN Volume Controller does not evaluate this attribute and therefore no action is taken when it is changed. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
ResetCapability	Uint16		If enabled, the ComputerSystem can be reset through hardware (the power and reset buttons). If disabled, hardware reset is not allowed. **Code** **Semantics** **1** Other **2** Unknown **3** Disabled **4** Enabled **5** Not Imple-mented
Roles	String[]	Write(TRUE)	Unsupported property.
Status	String	MaxLen(10), Deprecated(CIM_Managed SystemElement. OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to ″Other.″

Table 21. IBMTSSVC_IOGroup properties (continued)

Property	Type	Qualifier	Description
TimeOfLastState Change	Date-time		Unsupported property.

IBMTSSVC_IOGroupSet

The IBMTSSVC_IOGroup class extends the CIM_ExtraCapacitySet class.

Properties

An IOGroupSet defines an interface for a set of Volumes. All Nodes and Volumes are associated with exactly one IOGroupSet. The read and write cache provided by a node is duplicated for redundancy. When I/O is performed to a Volume, the node that processes the I/O will duplicate the data on the Partner node in the IOGroupSet. This class represents the set aspect of an I/O group, wheras IOGroup represents the system aspect.

The IBMTSSVC_IOGroupSet class extends the CIM_ExtraCapacitySet class and has the properties shown in Table 22.

Table 22. IBMTSSVC_IOGroupSet properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	A short textual description (one-line string) of the object.
Description	String		Provides a textual description of the object.
ElementName	String	Required(TRUE)	The user-friendly name for this instance of RedundancySet. (Note: InstanceName does not have to be unique within a namespace.)
InstanceID	String		Within the scope of the instantiating Namespace, InstanceID opaquely and uniquely identifies an instance of this class.
LoadBalancedSet	Boolean		Indicates whether load balancing is supported by the ExtraCapacitySet.

Table 22. IBMTSSVC_IOGroupSet properties (continued)

Property	Type	Qualifier	Description
MaxNumber Supported	Uint32		MaxNumber Supported indicates the largest number of elements that can participate in the ExtraCapacitySet. A value of 0 indicates there is no limit on the number of elements.
MinNumberNeeded	Uint32	MinValue(1)	MinNumberNeeded indicates the smallest number of elements that must be operational in order to function. For example, in an N+1 redundancy relationship, the MinNumberNeeded property is set equal to N.
RedundancyStatus	Uint16		RedundancyStatus provides information on the state of the RedundancySet. **Code** Semantics **0** Unknown **1** Other **2** Fully Redundant **3** Degraded Redundancy **4** Redundancy Lost

IBMTSSVC_Job

The IBMTSSVC_Job class extends the CIM_ConcreteJob class.

Properties

The IBMTSSVC_Job class is used to monitor the asynchronous commands for format, migration, or copy operations on the device. The IBMTSSVC_Job class extends the CIM_ConcreteJob class and has the properties shown in Table 23 on page 120.

Table 23. IBMTSSVC_Job properties

Property	Type	Qualifier	Description
InstanceID	String		Opaquely identifies a unique instance of ConcreteJob. The InstanceID must be unique within a namespace. To ensure this, the value of InstanceID should be constructed in the following manner: \n(Vendor ID)(ID) \n.
Caption	String	MaxLen(64)	Unsupported property.
DeleteOnCompletion	Boolean	Write(TRUE)	Indicates whether or not the job should be automatically deleted upon completion. If this property is set to false and the job completes, then the extrinsic method DeleteInstance must be used to delete the job.
Description	String		Provides a textual description of the object.
ElapsedTime	Date-time		Unsupported property.
ElementName	String		Unsupported property.
ErrorCode	Uint16	ModelCorrespondence (CIM_Job.ErrorDescription)	Unsupported property.
ErrorDescription	String	ModelCorrespondence (CIM_Job.ErrorCode)	Unsupported property.
InstallDate	Date-time		Unsupported property.
JobRunTimes	Uint32	Write(TRUE)	

Table 23. IBMTSSVC_Job properties (continued)

Property	Type	Qualifier	Description
JobState	Uint16		Integer enumeration that indicates the operational state of a Job. **Code** **Semantics** **2** New **3** Starting **4** Running **5** Suspended **6** Shutting Down **7** Completed **8** Terminated **9** Killed **10** Exception **11** Service **12..32767** DMTF Reserved **32768..65535** Vendor Reserved
JobStatus	String	ModelCorrespondence (ManagedSystemElement. OperationalStatus)	Unsupported property.
LocalOrUtcTime	Uint16	Write(TRUE)	**Code** **Semantics** **1** Local Time **2** UTC Time
Name	String	Required(TRUE), MaxLen(1024)	The user-friendly name for this instance of Job.
Notify	String	Write(TRUE)	Unsupported property.

Table 23. IBMTSSVC_Job properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. Various health and operational statuses are defined. Many of the enumeration's values are self-explanatory. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherRecoveryAction	String	ModelCorrespondence (CIM_Job.RecoveryAction)	Describes the recovery action when the instance's RecoveryAction property is set to ″Other.″
Owner	String	ModelCorrespondence (CIM_OwningJobElement)	Unsupported property.
PercentComplete	Uint16	MaxValue(101), MinValue(0), Units(Percent)	The percentage of the job that has completed at the time that this value is requested.
Priority	Uint32	Write(TRUE)	Unsupported property.

Table 23. IBMTSSVC_Job properties (continued)

Property	Type	Qualifier	Description
RecoveryAction	Uint16	ModelCorrespondence (CIM_Job.OtherRecoveryAction)	Describes the recovery action to be taken for an unsuccessfully run Job. **Code** **Semantics** **0** Unknown **1** Other **2** Do Not Continue **3** Continue With Next Job **4** Re-run Job **5** Run Recovery Job
RunDay	Uint8	Write(TRUE), MinValue(-31), MaxValue(31), ModelCorrespondence (CIM_Job.RunMonth CIM_Job.RunDayOfWeek CIM_Job.RunStartInterval)	
RunDayOfWeek	Uint8	Write(TRUE), ModelCorrespondence (CIM_Job.RunMonth CIM_Job.RunDay CIM_Job.RunStartInterval)	**Code** **Semantics** **-7** -Saturday **-6** -Friday **-5** -Thursday **-4** -Wednesday **-3** -Tuesday **-2** -Monday **-1** -Sunday **0** ExactDayOf Month **1** Sunday **2** Monday **3** Tuesday **4** Wednesday **5** Thursday **6** Friday **7** Saturday
RunMonth	Uint8	Write(TRUE), ModelCorrespondence (CIM_Job.RunDay CIM_Job.RunDayOfWeek CIM_Job.RunStartInterval)	**Code** **Semantics** **0** January **1** February **2** March **3** April **4** May **5** June **6** July **7** August **8** September **9** October **10** November **11** December
RunStartInterval	Date-time	Write(TRUE), ModelCorrespondence (CIM_Job.RunMonth CIM_Job.RunDay CIM_Job.RunDayOfWeek CIM_Job.RunStartInterval)	

Table 23. IBMTSSVC_Job properties (continued)

Property	Type	Qualifier	Description
ScheduledStartTime	Date-time	Deprecated (CIM_Job.RunMonth CIM_Job.RunDay CIM_Job.RunDayOfWeek CIM_Job.RunStartInterval), Write(TRUE)	Unsupported property.
StartTime	Date-time		Unsupported property.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystemElement. OperationalStatus)	This property is deprecated in lieu of OperationalStatus.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystemElement. OperationalStatus)	Describes the status - used when the OperationalStatus property is set to 1 (″Other″).
TimeOfLast StateChange	Date-time		Unsupported Property.
TimeSubmitted	Date-time		Unsupported property.
UntilTime	Date-time	Write(TRUE), ModelCorrespondence (CIM_Job.LocalOrUtcTime)	Unsupported property.

IBMTSSVC_MessageLog

The IBMTSSVC_MessageLog class extends the CIM_MessageLog class.

Properties

The IBMTSSVC_MessageLog class is The IBMTSSVC_MessageLog class extends the CIM_MessageLog class and has the properties shown in Table 24 on page 125

Table 24. IBMTSSVC_MessageLog properties

Property	Type	Qualifier	Description
Capabilities	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_MessageLog.Capabilities Descriptions)	An array of integers indicating the Log capabilities. **Code** **Semantics** **0** Unknown **1** Other **2** Write Record Supported **3** Delete Record Supported **4** Can Move Backward in Log **5** Freeze Log Supported **6** Clear Log Supported **7** Supports Addressing by Ordinal Record Number **8** Variable Length Records Supported **9** Variable Formats for Records **10** Can Flag Records for Overwrite
Capabilities Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_MessageLog.Capabilities)	An array of free-form strings providing more detailed explanations for any of the Log features indicated in the Capabilities array. Note that each entry of this array is related to the entry in the Capabilities array that is located at the same index.
Caption	String	MaxLen(64)	The Caption property is a short textual description (one-line String) of the object.

Table 24. IBMTSSVC_MessageLog properties (continued)

Property	Type	Qualifier	Description
CharacterSet	Uint16		An enumeration describing the character set used to record data in the individual Log entries. **Code** **Semantics** **0** Unknown **1** Other **2** ASCII **3** Unicode **4** ISO2022 **5** ISO8859 **6** Extended UNIX Code **7** UTF-8 **8** UCS-2 **9** Bitmapped Data **10** OctetString **11** Defined by Individual Records
CreationClassName	String	MaxLen(256)	CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
CurrentNumberOf Records	Uint64	Gauge(TRUE)	Current number of entries (records) in the Log.
Description	String		The Description property provides a textual description of the object.
ElementName	String		A user-friendly name for the object. This property allows each instance to define a user-friendly name, in addition to its key properties/identity data, and description information. \n.

Table 24. IBMTSSVC_MessageLog properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
HeaderFormat	String		If the SizeOfHeader property is non-zero, this property describes the structure and format of the Log header. It is a free-form String. If the SizeOfHeader property is 0, then the information in this property is undefined.

Table 24. IBMTSSVC_MessageLog properties (continued)

Property	Type	Qualifier	Description
InstallDate	Date-time		A Datetime value indicating when the object was installed. A lack of a value does not indicate that the object is not installed.
IsFrozen	Boolean		Boolean indicating that the Log is currently frozen and modifications are not allowed.
LastChange	Uint16		An enumeration describing the last change to the MessageLog. **Code** **Semantics** **0** Unknown **1** Add **2** Delete **3** Modify **4** Log Cleared
MaxLogSize	Uint64	Units(Bytes)	The maximum size, in bytes, to which the Log can grow. If there is no maximum, then MaxLogSize should be set to 0.
MaxNumberOf Records	Uint64		Maximum number of records that can be captured in the Log. If undefined, a value of zero should be specified.
MaxRecordSize	Uint64	Units(Bytes)	Maximum size, in bytes, to which an individual Log entry (record) can grow - if the Capabilities array includes a value of 7 (″Variable Length Records Supported″). If the Capabilities array does not include a 7, then the Log only supports fixed length entries.
Name	String	MaxLen(256)	The inherited Name serves as part of the key (a unique identifier) for the MessageLog instance.

Table 24. IBMTSSVC_MessageLog properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
OtherPolicy Description	String	ModelCorrespondence (CIM_MessageLog. OverwritePolicy)	When the OverwritePolicy specifies a value of 1 (″Other″), the Log's behavior can be explained by this property. If OverwritePolicy is not 1, then this property's contents are undefined.

Table 24. IBMTSSVC_MessageLog properties (continued)

Property	Type	Qualifier	Description
OverwritePolicy	Uint16	ModelCorrespondence (CIM_MessageLog. OtherPolicyDescription CIM_MessageLog.TimeWhen Outdated CIM_MessageLog PercentageNearFull)	An enumeration describing the behavior of the Log when it becomes full or near full. **Code** **Semantics** **0** Unknown **1** Other **2** Wraps When Full **3** Clear When Near Full **4** Overwrite Outdated When Needed **5** Remove Outdated Records **6** Overwrite Specific Records **7** Never Overwrite
PercentageNearFull	Uint8	Units(Percent), ModelCorrespondence (CIM_MessageLog. OverwritePolicy)	If the OverwritePolicy is based on clearing records when the Log is near full (value=3), this property defines the record capacity (in percentage) that is considered to be 'near full.'
RecordHeader Format	String		If the SizeOfRecordHeader property is non-zero, this property describes the structure and format of the record headers. It is a free-form string. If the SizeOfRecordHeader property is 0, then the information in this property is undefined.
RecordLast Changed	Uint64		When a change is made to the Log, the record number that was modified is captured.

Table 24. IBMTSSVC_MessageLog properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. This property is provided to compare Requested and current Enabledstatus. Note that the SAN Volume Controller does not evaluate this attribute and therefore no action is taken when it is changed. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
SizeOfHeader	Uint64	Units(Bytes)	The size of the Log header, in bytes, if one is present. If there is no Log header, then this property should be set to 0.
SizeOfRecordHeader	Uint64	Units(Bytes)	The size of the header for the Log's individual entries, in bytes, if record headers are defined. If there are no record headers, then this property should be set to 0.
Status	String	MaxLen(10), Deprecated(CIM_Managed SystemElement.Operational Status)	A string indicating the current status of the object. This property is deprecated in lieu of OperationalStatus, which includes the same semantics in its enumeration.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).

Table 24. IBMTSSVC_MessageLog properties (continued)

Property	Type	Qualifier	Description
TimeOfLastChange	Date-time		When a change is made to the Log, the date/time of that modification is captured. This property could be used to event against any update to the MessageLog.
TimeOfLastState Change	Date-time		Unsupported property.
TimeWhenOutdated	Date-time	ModelCorrespondence (CIM_MessageLog. OverwritePolicy)	If the OverwritePolicy is based on 'outdated' records (values 4 or 5), this property defines when a Log entry is considered to be outdated, either by time interval or at a specific date and time.

IBMTSSVC_Node

The IBMTSSVC_Node class represents a single SAN Volume Controller node that is part of a cluster.

Properties

A cluster can contain up to eight nodes or four node pairs. The IBMTSSVC_Node class extends the CIM_ComputerSystem class and has the properties shown in Table 25.

Table 25. IBMTSSVC_Node properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	The Caption property is a short textual description (one-line string) of the object.
CreationClassName	String	MaxLen(256)	Indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.

Table 25. IBMTSSVC_Node properties (continued)

Property	Type	Qualifier	Description
Dedicated	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherDedicatedDescriptions)	Enumeration indicating whether the ComputerSystem is a special-purpose System (dedicated to a particular use), versus being 'general purpose.' SAN Volume Controller is a dedicated storage device and will return {3,15} (″Storage,″ ″Block Server″). **Code** **Semantics** **0** Not Dedicated **1** Unknown **2** Other **3** Storage **4** Router **5** Switch **6** Layer 3 Switch **7** Central Office Switch **8** Hub **9** Access Server **10** Firewall **11** Print **12** I/O **13** Web Caching **14** Management **15** Block Server **16** File Server **17** Mobile User Device **18** Repeater **19** Bridge/ Extender **20** Gateway
Description	String		The Description property provides a textual description of the object.
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	The node's user-friendly name

Table 25. IBMTSSVC_Node properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. OtherIdentifyingInfo)	An array of free-form strings providing explanations and details behind the entries in the OtherIdentifyingInfo array. Note that each entry of this array is related to the entry in OtherIdentifyingInfo that is located at the same index.
InstallDate	Date-time		Unsupported property.

Table 25. IBMTSSVC_Node properties (continued)

Property	Type	Qualifier	Description
IsConfigNode	Boolean		True, if this node is the config node of its cluster.
Name	String	MaxLen(256)	The label by which the object is known. Format: cluster_ip:object_id
NameFormat	String	MaxLen(64)	Identifies how the ComputerSystem Name is generated. SAN Volume Controller returns the node's id as Name, therefore this attribute is set to ″Other.″
NativeStatus	Uint16		The node's native operational status. **Code** **Semantics** **0** Offline **1** Online **2** Pending **3** Adding **4** Deleting
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The node's operational status. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherDedicated Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. Dedicated)	A string describing how or why the system is dedicated when the Dedicated array includes the value 2, (Other″).

Table 25. IBMTSSVC_Node properties (continued)

Property	Type	Qualifier	Description
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
OtherIdentifyingInfo	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_ComputerSystem. IdentifyingDescriptions), Expensive(TRUE)	The node's ″Redundancy Group ID,″ ″RedundancyGroup Name,″ ″Partner Node Name,″ ″Partner Node ID″ and ″WWWN.″
PowerManagement Capabilities	Uint16[]	Deprecated (CIM_PowerManagement Capabilities.PowerCapabilities)	An enumerated array describing the power management capabilities of the ComputerSystem. The use of this property has been deprecated. Instead, the PowerCapabilites property in an associated PowerManagement Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE),	Unsupported property.
PrimaryOwner Name	String	MaxLen(64), Write(TRUE),	Unsupported property.

Table 25. IBMTSSVC_Node properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. Note that SAN Volume Controller does not evaluate this attribute and therefore no action is taken when it is changed. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
ResetCapability	Uint16		If enabled (value = 4), the ComputerSystem can be reset through hardware (the power and reset buttons). If disabled (value = 3), hardware reset is not allowed. **Code** **Semantics** **1** Other **2** Unknown **3** Disabled **4** Enabled **5** Not Implemented
Roles	String[]	Write(TRUE)	Unsupported property.
Status	String	MaxLen(10), Deprecated(CIM_Managed SystemElement. OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
TimeOfLastState Change	Date-time		Unsupported property.

IBMTSSVC_NodeVPD

The IBMTSSVC_NodeVPD class contains the vital product data of the corresponding instance.

Properties

The IBMTSSVC_NodeVPD class contains the vital product data (VPD) of the corresponding SAN Volume Controller IBMTSSVC_Node instance.The IBMTSSVC_NodeVPD class extends the CIM_SettingData class and has the properties shown in Table 26.

Table 26. IBMTSSVC_NodeVPD properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Unsupported Property.
Description	String		Provides a textual description of the object
ElementName	String	Required(TRUE)	Unsupported Property.
EthernetIP	String		EthernetIP
FrontPanetID	String		FrontPanelID of the node.
InstanceID	String	Key	Opaquely identifies a unique instance of SettingData. The InstanceID must be unique within a namespace. To ensure this, the value of InstanceID should be constructed in the following manner: \n(Vendor ID)(ID) \n
NodeVPD	String[]		Specifies the VPD of the SAN Volume Controller.

IBMTSSVC_PrimordialStoragePool

The IBMTSSVC_PrimordialStoragePool class extends the CIM_StoragePool class.

Properties

The IBMTSSVC_PrimordialStoragePool class extends the CIM_StoragePool class and has the properties shown in Table 27.

Table 27. IBMTSSVC_PrimordialStoragePool properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(15)	A short textual description (one-line string) of the object.
Description	String		Provides a textual description of the object.
ElementName	String	MaxLen(15)	The pool's user-friendly name.
InstallDate	Date-time		Unsupported Property.

Table 27. IBMTSSVC_PrimordialStoragePool properties (continued)

Property	Type	Qualifier	Description
InstanceID	String		The label by which the object is known. Format: cluster_ip:P:object_id
Name	String	MaxLen(1024)	The pool's globally unique id. The ID format is (Vendor)(id), in case of SAN Volume Controller IBMTSSVC(id).
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Unsupported, always reported as ″Unknown.″ **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
PoolID	String	Required(TRUE), MaxLen(256)	This is numeric and only unique in terms of the hosting SAN Volume Controller cluster.

Table 27. IBMTSSVC_PrimordialStoragePool properties (continued)

Property	Type	Qualifier	Description
Primordial	Boolean		If true, ″Primordial″ indicates that the containing System does not have the ability to create or delete this operational element. Higher-level StoragePools may be assembled using the Component or AllocatedFromStorage Pool associations. Although the higher-level abstractions can be created and deleted, the most basic (primordial) hardware-based StoragePools cannot. They are physically realized as part of the System, or are actually managed by some other System and imported as if they were physically realized.
RemainingManaged Space	Uint64	Units(Bytes), ModelCorrespondence (StoragePool.TotalManaged Space AllocatedFromStorage Pool.SpaceConsumed), Required(TRUE)	The remaining amount of raw storage (in bytes) from the TotalManagedSpace of this StoragePool.
Status	String	MaxLen(10), Deprecated(CIM_Managed SystemElement.Operationa lStatus)	Deprecated property - set to ″Unknown.″ Look to OperationalStatus for status information
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
TotalManagedSpace	Uint64	Units(Bytes), ModelCorrespondence (StoragePool.Remaining ManagedSpace)	The total amount of raw storage (in bytes) managed by this StoragePool.

IBMTSSVC_Privilege

IBMTSSVC_Privilege class extends the CIM_AuthorizedPrivilege class.

Properties

The IBMTSSVC_Privilege class extends the CIM_AuthorizedPrivilege class and has the properties shown in Table 28 on page 141.

Table 28. IBMTSSVC_Privilege properties

Property	Type	Qualifier	Description
Activities	Uint16[]	ModelCorrespondence (CIM_Privilege.Activity Qualifiers), ArrayType(Indexed)	An array of string values indicating the activities that are granted or denied. These activities apply to all entities specified in the ActivityQualifiers array. **Code** **Semantics** **0** Unknown **1** Other **2** Create **3** Delete **4** Read **5** Write **6** Execute **7..** DMTF Reserved
ActivityQualifiers	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_Privilege.Activities CIM_Privilege.QualifierFormats)	Unsupported Property
Caption	String	MaxLen(64)	Unsupported Property
Description	String		Provides a textual description of the object.
ElementName	String		Unsupported Property
InstanceID	String		Opaquely identifies a unique instance of Privilege. The InstanceID must be unique within a namespace. To ensure this, the value of InstanceID should be constructed in the following manner: \n(Vendor/Admin ID):(ID) \n (Vendor/Admin ID) MUST include a copyrighted, trademarked or otherwise unique name that is owned by the business entity, or a registered ID.
PrivilegeGranted	Boolean		Boolean indicating whether this Privilege grants (TRUE) or denies (FALSE) permission. The default is to grant permission.

Table 28. IBMTSSVC_Privilege properties (continued)

Property	Type	Qualifier	Description
QualifierFormats	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_Privilege.Activity Qualifiers)	Unsupported Property. **Code** **Semantics** **2** Class Name **3** <Class.> Property **4** <Class.> Method **5** Object Reference **6** Namespace **7** URL **8** Directory/File Name **9** Command Line Instruction **..15999** DMTF Reserved **16000..** Vendor Reserved

IBMTSSVC_Product

The IBMTSSVC_Product class extends the CIM_Product class.

Properties

The IBMTSSVC_Product class represents a SAN Volume Controller unit and contains the ordering and version information. The Product instance aggregates the PhysicalElements, software, services and other components of the SAN Volume Controller.

The IBMTSSVC_Product class extends the CIM_Product class and has the properties shown in Table 29.

Table 29. IBMTSSVC_Product properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description.
Description	String		Provides a textual description of the object
ElementName	String		Specifies the name of the Product
IdentifyingNumber	String	MaxLen(64)	Specifies the Product identification, such as a serial number on software, a die-number on a hardware chip, or (for noncommercial Products) a project number

Table 29. IBMTSSVC_Product properties (continued)

Property	Type	Qualifier	Description
Name	String	MaxLen(256), ModelCorrespondence (CIM.DMTF|PRS_Product. ProductName)	Specifies the Product name
SKUNumber	String	MaxLen(64)	Unsupported property
Vendor	String	MaxLen(256), ModelCorrespondence (CIM.DMTF|PRS_ Product.Vendor)	Specifies the name of the supplier, manufacturer, or reseller of the Product. Corresponds to the Vendor property in the Product object in the DMTF Solution Exchange Standard.
Version	String	MaxLen(64), ModelCorrespondence (CIM.DMTF|PRS_ Product.Version)	Specifies the version of the PhysicalElement.
WarrantyDuration	Uint32	Units(Days), ModelCorrespondence (CIM_Product.Warranty StartDate)	Unsupported property
WarrantyStartDate	Date-time	ModelCorrespondence (CIM_Product.Warranty Duration)	Unsupported property

IBMTSSVC_Provider

The IBMTSSVC_Provider class extends the CIM_Provider class.

Properties

The IBMTSSVC_Provider class has the properties shown in Table 30.

Table 30. IBMTSSVC_Provider properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64), ReadRole(None)	Short textual description (one-line string) of the object.
CreationClassName	String	MaxLen(256)	Indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String	ReadRole(None)	Textual description of the object.

Table 30. IBMTSSVC_Provider properties (continued)

Property	Type	Qualifier	Description
ElementName	String	ReadRole(None)	A user-friendly name for the object. This property allows each instance to define a user-friendly name, in addition to its key properties/identity data and description information. \n.
EnabledDefault	uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
Handle	String	Required(TRUE)	An implementation-specific string that identifies the handle to the provider.

Table 30. IBMTSSVC_Provider properties (continued)

Property	Type	Qualifier	Description
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(256)	A human-readable name that uniquely identifies the provider within a system.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The operational status of the service. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabled State	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledStatus property is set to 1 (″Other″). This property MUST be set to NULL when EnabledStatus is any value other than 1.
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property

Table 30. IBMTSSVC_Provider properties (continued)

Property	Type	Qualifier	Description
RequestedState	uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. **Code Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Started	Boolean		Indicates if this service is started.
StartMode	String	MaxLen(10), Deprecated (CIM_Service.Enabled Default)	Indicates if this service is started manually or automatically.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
SystemCreation ClassName	String	Propagated (CIM_System.Creation ClassName), MaxLen(256)	The scoping system's creation class name.
SystemName	String	Propagated (CIM_System.Name), MaxLen(256)	The scoping system's name.
TimeOfLast StateChange	Date-time		Unsupported property

IBMTSSVC_RegisteredProfile

The IBMTSSVC_RegisteredProfile class represents a SAN Volume Controller unit and contains the ordering and version information.

Properties

The Product instance aggregates the PhysicalElements, software, services and other components of the SAN Volume Controller.

A RegisteredProfile describes a set of Common Information Model (CIM) Schema classes with required properties and/or methods, necessary to manage a real-world entity or to support a usage scenario, in an interoperable fashion.

RegisteredProfiles can be defined by the DMTF or other standards organizations. Note that this class should not be confused with CIM_Profile, which collects SettingData instances to be applied as a ″configuration profile″ for an element.

The IBMTSSVC_RegisteredProfile class extends the CIM_RegisteredProfile class and has the properties shown in Table 31.

Table 31. IBMTSSVC_RegisteredProfile properties

Property	Type	Qualifier	Description
AdvertiseType Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_RegisteredProfile. AdvertiseTypes)	A free-form string providing additional information related to the AdvertiseType. A description MUST be provided when the AdvertiseType is 1 (″Other″).
AdvertiseTypes	uint16[]	Required(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_RegisteredProfile. AdvertiseTypeDescriptions)	Signifies the advertisement for the profile information. It is used by the advertising services of the WBEM infrastructure to determine what should be advertised, via what mechanisms. The property is an array so that the profile may be advertised using several mechanisms. Note: If this property is null/uninitialized, this is equivalent to specifying the value 2, ″Not Advertised.″ **Code** **Semantics** **1** Other **2** Not Advertised **3** SLP
Caption	String	MaxLen(64)	A short textual description (one-line string) of the object.
Description	String		Provides a textual description of the object.
ElementName	String		A user-friendly name for the object. This property allows each instance to define a user-friendly name, in addition to its key properties/identity data, and description information.

Table 31. IBMTSSVC_RegisteredProfile properties (continued)

Property	Type	Qualifier	Description
InstanceID	String		Opaquely and uniquely identifies an instance of this class. In order to ensure this, the value of InstanceID SHOULD be constructed using the following 'preferred' algorithm: \n <OrgID>:<LocalID> \nWhere <OrgID> and <LocalID> are separated by a colon ':', and where <OrgID> MUST include a copyrighted, trademarked or otherwise unique name that is owned by the business entity creating/defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority.
OtherRegistered Organization	String	MaxLen(256), ModelCorrespondence (CIM_RegisteredProfile. RegisteredOrganization)	A free-form string providing a description of the organization when 1, ″Other,″ is specified for the Registered Organization.
RegisteredName	String	Required(TRUE), MaxLen(256)	A string to identify this RegisteredProfile. It is the responsibility of the defining organization to ensure that the profile's name is unique within the scope of the organization.

Table 31. IBMTSSVC_RegisteredProfile properties (continued)

Property	Type	Qualifier	Description
Registered Organization	uint16	Required(TRUE), ModelCorrespondence (CIM_RegisteredProfile. OtherRegisteredOrganization)	The organization that defines this profile. **Code** **Semantics** **1** Other **2** DMTF **3** CompTIA **4** Consortium for Service Innovation **5** FAST **6** GGF **7** INTAP **8** itSMF **9** NAC **10** Northwest Energy Efficiency Alliance **11** SNIA **12** TM Forum **13** The Open Group **14** ANSI **15** IEEE **16** IETF **17** INCITS **18** ISO **19** W3C
Registered Version	String	Required(TRUE)	The version of this profile. The String representing the version MUST be in the form: \n M + ″.″ + N + ″.″ + U \nWhere: \nM - The major version (in numeric form) describing the profile's creation or last modification. \nN.

IBMTSSVC_RegisteredSubProfile

The IBMTSSVC_RegisteredProfile class extends the CIM_RegisteredSubProfile class.

Properties

A RegisteredSubProfile subclasses RegisteredProfile to indicate that a scoping profile is required to provide context. The latter is specified by the mandatory association, SubProfileRequiresProfile.

The IBMTSSVC_RegisteredProfile class has the properties shown in Table 32 on page 150.

Table 32. IBMTSSVC_RegisteredSubProfile properties

Property	Type	Qualifier	Description
AdvertiseType Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_RegisteredProfile. AdvertiseTypes)	A free-form string providing additional information related to the AdvertiseType. A description MUST be provided when the AdvertiseType is 1 (″Other.″).
AdvertiseTypes	Uint16[]	Required(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_RegisteredProfile. AdvertiseTypeDescriptions)	Signifies the advertisement for the profile information. It is used by the advertising services of the WBEM infrastructure to determine what should be advertised, via what mechanisms. **Code** **Semantics** **1** Other **2** Not Advertised **3** SLP
Caption	String	MaxLen(64)	A short textual description (one-line string) of the object.
Description	String		Provides a textual description of the object.
ElementName	String		A user-friendly name for the object.

Table 32. IBMTSSVC_RegisteredSubProfile properties (continued)

Property	Type	Qualifier	Description
InstanceID	String		Opaquely and uniquely identifies an instance of this class. In order to ensure this, the value of InstanceID should be constructed using the following 'preferred' algorithm: \n <OrgID>:<LocalID> \nWhere <OrgID> and <LocalID> are separated by a colon ':', and where <OrgID> MUST include a copyrighted, trademarked or otherwise unique name that is owned by the business entity creating/defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority.
OtherRegistered Organization	String	MaxLen(256), ModelCorrespondence (CIM_RegisteredProfile. RegisteredOrganization)	A free-form string providing a description of the organization when 1 (″Other,″) is specified for the Registered Organization.
RegisteredName	String	Required(TRUE), MaxLen(256)	A string to identify this RegisteredProfile.

Table 32. IBMTSSVC_RegisteredSubProfile properties (continued)

Property	Type	Qualifier	Description
Registered Organization	Uint16	Required(TRUE), ModelCorrespondence (CIM_RegisteredProfile. OtherRegisteredOrganization)	The organization that defines this profile. **Code** **Semantics** **1** Other **2** DMTF **3** CompTIA **4** Consortium for Service Innovation **5** FAST **6** GGF **7** INTAP **8** itSMF **9** NAC **10** Northwest Energy Efficiency Alliance **11** SNIA **12** TM Forum **13** The Open Group **14** ANSI **15** IEEE **16** IETF **17** INCITS **18** ISO **19** W3C
RegisteredVersion	String	Required(TRUE)	The version of this profile. The string representing the version MUST be in the form: \n M + ″.″ + N + ″.″ + U \nWhere: \nM - The major version (in numeric form) describing the profile's creation or last modification.

IBMTSSVC_RemoteCluster

The IBMTSSVC_RemoteCluster class represents a separate SAN Volume Controller cluster connected through the fibre-channel network to the local cluster on which a synchronous copy partnership has been established.

Properties

The IBMTSSVC_RemoteCluster class extends the IBMTSSVC_AbstractCluster class and has the properties shown in Table 33 on page 153:

Table 33. IBMTSSVC_RemoteCluster properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
Description	String		Provides a textual description of the object
ElementName	String		Specifies the name of the cluster
InstallDate	Date-time		A datetime value indicating when the object was installed. A lack of a value does not indicate that the object is not installed.
IP	String		The IP address of the remote cluster.
Name	String	MaxLen(256)	Labels the new object
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode **..** DMTF Reserved **0x8000..** Vendor Reserved
Partnership Bandwidth	String		The bandwith used for this partnership.

Table 33. IBMTSSVC_RemoteCluster properties (continued)

Property	Type	Qualifier	Description
PartnershipStatus	String		The status of the remote cluster partnership. May be either ″Fully_Configured,″ ″Partly_Configured,″ or ″Offline.″
ServiceIP	String		The service IP address of the remote cluster.
Status	String	Deprecated (CIM_ManagedSystem Element.OperationalStatus), MaxLen(10)	A string indicating the current status of the object. This property is deprecated in lieu of OperationalStatus, which includes the same semantics in its enumeration.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	Strings describing the various OperationalStatus array values.
SystemName	String	MaxLen(256)	The label by which the object is known.

IBMTSSVC_RemoteServiceAccessPoint

RemoteServiceAccessPoint describes access and/or addressing information for a remote connection, that is known to a local network element.

Properties

The IBMTSSVC_RemoteServiceAccessPoint class extends the CIM_RemoteServiceAccessPoint class and has the properties shown in Table 34.

Table 34. IBMTSSVC_RemoteServiceAccessPoint properties

Property	Type	Qualifier	Description
AccessInfo	String	ModelCorrespondence (CIM_RemoteService AccessPoint.InfoFormat)	Access and/or addressing information for a remote connection. This can be a host name, network address or similar information.
Caption	String	MaxLen(64)	A short textual description (one- line string) of the object.
ConsoleIP	String	Expensive(TRUE)	The IP address of the management console.
ConsolePort	String	Expensive(TRUE)	The port address of the management console.

Table 34. IBMTSSVC_RemoteServiceAccessPoint properties (continued)

Property	Type	Qualifier	Description
CreationClass Name	String	MaxLen(256)	CreationClass Name indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		Provides a textual description of the object.
ElementName	String		A user-friendly name for the object. This property allows each instance to define a user-friendly name, in addition to its key properties/identity data, and description information.
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's Enabled State. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 34. IBMTSSVC_RemoteServiceAccessPoint properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	EnabledState is an integer enumeration that indicates the enabled/disabled states of an element. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 34. IBMTSSVC_RemoteServiceAccessPoint properties (continued)

Property	Type	Qualifier	Description
InfoFormat	Uint16	ModelCorrespondence (CIM_RemoteService AccessPoint.Other InfoFormatDescription)	An enumerated integer describing the format and interpretation of the AccessInfo property. **Code** **Semantics** **1** Other **2** Host Name **3** IPv4 Address **4** IPv6 Address **5** IPX Address **6** DECnet Address **7** SNA Address **8** Autonomous System Number **9** MPLS Label **10** IPv4 Subnet Address **11** IPv6 Subnet Address **12** IPv4 Address Range **13** IPv6 Address Range **100** Dial String **101** Ethernet Address **102** Token Ring Address **103** ATM Address **104** Frame Relay Address **200** URL **201** FQDN **202** User FQDN **203** DER ASN1 DN **204** DER ASN1 GN **205** Key ID **..** DMTF Reserved **32768..65535** Vendor Reserved
InstallDate	Date-time		A datetime value indicating when the object was installed. A lack of a value does not indicate that the object is not installed.

Table 34. IBMTSSVC_RemoteServiceAccessPoint properties (continued)

Property	Type	Qualifier	Description
Name	String	MaxLen(256)	The Name property uniquely identifies the ServiceAccess Point and provides an indication of the functionality that is managed. This functionality is described in more detail in the object's Description property.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode **..** DMTF Reserved **0x8000..** Vendor Reserved
OtherEnabled State	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property must be set to NULL when EnabledState is any value other than 1.
OtherInfoFormat Description	String	ModelCorrespondence (CIM_RemoteService AccessPoint.InfoFormat)	Describes the format when the property InfoFormat is set to 1 (″Other″).

Table 34. IBMTSSVC_RemoteServiceAccessPoint properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicating the last requested or desired state for the element. The actual state of the element is represented by EnabledState. This property is provided to compare the last requested and current enabled/disabled states. Note that when EnabledState is set to 5 (″Not Applicable″), then this property has no meaning. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10** Reboot **11** Reset **..** DMTF Reserved **32768..65535** Vendor Reserved
Status	String	Deprecated (CIM_ManagedSystem Element.OperationalStatus), MaxLen(10)	A string indicating the current status of the object. This property is deprecated in lieu of OperationalStatus, which includes the same semantics in its enumeration.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	Strings describing the various OperationalStatus array values. For example, if ″Stopping″ is the value assigned to OperationalStatus, then this property may contain an explanation as to why an object is being stopped.
SystemCreation ClassName	String	Propagated (CIM_System.Creation ClassName), MaxLen(256)	The scoping System's CreationClassName.

Table 34. IBMTSSVC_RemoteServiceAccessPoint properties (continued)

Property	Type	Qualifier	Description
SystemName	String	Propagated (CIM_System.Name), MaxLen(256)	The scoping System's Name.
TimeOfLast StateChange	Date-time		The date/time when the element's EnabledState last changed. If the state of the element has not changed and this property is populated, then it MUST be set to a 0 interval value. If a state change was requested, but rejected or not yet processed, the property must not be updated.

IBMTSSVC_RemoteVolume

The IBMTSSVC_RemoteVolume class represents a remote volume in a synchronous copy relationship.

Properties

The IBMTSSVC_RemoteVolume class extends the CIM_LogicalElement class and has the properties shown in Table 35.

Table 35. IBMTSSVC_RemoteVolume properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	A short textual description (one- line string) of the object
ClusterID	String		The identifier of the remote volume's cluster
ClusterName	String		The name of the remote volume's cluster
Description	String		Provides a textual description of the object
ElementName	String		A user-friendly name for the object
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(1024)	The identifier of the remote volume

Table 35. IBMTSSVC_RemoteVolume properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	Indicates the current status of the element. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	A string indicating the current status of the object; reported as *Unknown*
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	Describes the status. Is used when the OperationalStatus property is set to 1 (″Other″).
SystemName	String		The IP of the scoping cluster.

IBMTSSVC_StorageCapabilities

IBMTSSVC_StorageCapabilities class is a subclass of Capabilities that defines the Capabilities of a StorageService or StoragePool.

Properties

For example, an instance of StorageCapabilities could be associated with either a StorageConfigurationService or StoragePool by using ElementCapabilities. The IBMTSSVC_StorageCapabilities class extends the CIM_StorageCapabilities class and has the properties shown in Table 36 on page 162.

Table 36. IBMTSSVC_StorageCapabilities properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	A short textual description (one-line String) of the object.
DataRedundancy Default	Uint16	MinValue(1), ModelCorrespondence (CIM_StorageCapabilities. DataRedundancyMax CIM_StorageCapabilities. DataRedundancyMin)	Describes the default number of complete copies of data that can be maintained.
DataRedundancy Max	Uint16	MinValue(1), ModelCorrespondence (CIM_StorageCapabilities. DataRedundancyMin CIM_StorageCapabilities. DataRedundancyDefault)	Describes the maximum number of complete copies of data that can be maintained.
DataRedundancy Min	Uint16	MinValue(1), ModelCorrespondence (CIM_StorageCapabilities. DataRedundancyMax CIM_StorageCapabilities. DataRedundancyDefault)	Describes the minimum number of complete copies of data that can be maintained.
DeltaReservation Default	Uint16	MinValue(0), MaxValue(100), Units(Percentage), ModelCorrespondence (CIM_StorageCapabilities. DeltaReservationMax CIM_StorageCapabilities. DeltaReservationMin)	Delta reservation is a number between 1 (1%) and a 100 (100%) that specifies how much space should be reserved by default in a replica for caching changes.
DeltaReservation Max	Uint16	MinValue(0), MaxValue(100), Units(Percentage), ModelCorrespondence (CIM_StorageCapabilities. DeltaReservationMin CIM_StorageCapabilities. DeltaReservationDefault)	A number between 1 (1%) and a 100 (100%) that specifies the maximum amount of space reserved in a replica for caching changes.
DeltaReservation Min	Uint16	MinValue(0), MaxValue(100), Units(Percentage), ModelCorrespondence (CIM_StorageCapabilities. DeltaReservationMax CIM_StorageCapabilities. DeltaReservationDefault)	A number between 1 (1%) and a 100 (100%) that specifies the minimum amount of space that should be reserved in a replica for caching changes.
Description	String		Provides a textual description of the object.
ElementName	String	Required(TRUE)	The user-friendly name for this instance of Capabilities.

Table 36. IBMTSSVC_StorageCapabilities properties (continued)

Property	Type	Qualifier	Description
ElementType	Uint16		Enumeration indicating the type of element to which this StorageCapabilities applies. **Code** **Semantics** **0** Unknown **1** Reserved **2** Any Type **3** StorageVolume **4** StorageExtent **5** StoragePool **6** Storage Configuration Service
InstanceID	String		Within the scope of the instantiating Namespace, InstanceID opaquely and uniquely identifies an instance of this class. In order to ensure this, the value of InstanceID should be constructed using the following algorithm: \n <OrgID>:<LocalID > \nWhere <OrgID> and <LocalID> are separated by a colon ':', and where <OrgID> MUST include a copyrighted, trademarked or otherwise unique name that is owned by the business entity creating/ defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority.
NoSinglePoint OfFailure	Boolean	ModelCorrespondence (CIM_StorageCapabilities. NoSinglePointOfFailure Default)	Indicates whether or not the associated element supports no single point of failure. Values are: FALSE = does not support no single point of failure, and TRUE = supports no single point of failure.
NoSinglePoint OfFailureDefault	Boolean	ModelCorrespondence (CIM_StorageCapabilities. NoSinglePointOfFailure)	Indicates the default value for the NoSinglePointOfFailure property.

Table 36. IBMTSSVC_StorageCapabilities properties (continued)

Property	Type	Qualifier	Description
Package RedundancyDefault	Uint16	ModelCorrespondence (CIM_StorageCapabilities. PackageRedundancyMin CIM_StorageCapabilities. PackageRedundancyMax)	Describes the default number of redundant packages that will be used.
Package RedundancyMax	Uint16	ModelCorrespondence (CIM_StorageCapabilities. PackageRedundancyMin CIM_StorageCapabilities. PackageRedundancyDefault)	Describes the maximum number of redundant packages that can be used.
Package RedundancyMin	Uint16	ModelCorrespondence (CIM_StorageCapabilities. PackageRedundancyMax CIM_StorageCapabilities. PackageRedundancyDefault)	Describes the minimum number of redundant packages that can be used.

IBMTSSVC_StorageConfigurationCapabilities

IBMTSSVC_StorageConfigurationCapabilities class is a subclass of Capabilities that defines the Capabilities of a StorageConfigurationService.

Properties

An instance of StorageConfigurationCapabilities is associated with a StorageConfigurationService using ElementCapabilities. The IBMTSSVC_StorageConfigurationCapabilities class extends the CIM_StorageConfigurationCapabilities class and has the properties shown in Table 37.

Table 37. IBMTSSVC_StorageConfigurationCapabilities properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	A short textual description (one-line string) of the object.
Description	String		Provides a textual description of the object.
ElementName	String	Required(TRUE)	The user-friendly name for this instance of Capabilities. In addition, the user-friendly name can be used as a index property for a search of query. (Note: Name does not have to be unique within a namespace.)

Table 37. IBMTSSVC_StorageConfigurationCapabilities properties (continued)

Property	Type	Qualifier	Description
Initial ReplicationState	uint16	ModelCorrespondence (CIM_StorageConfiguration Service.AttachReplica CIM_StorageConfiguration Service.CreateReplica)	Specifies which initial Replication State is supported by a particular provider. **Code** **Semantics** **2** Initialized **3** Prepared **4** Synchronized **..** DMTF Reserved **0x8000..0xFFFF** Vendor Specific
InstanceID	String		Within the scope of the instantiating Namespace, InstanceID opaquely and uniquely identifies an instance of this class. In order to ensure this, the value of InstanceID SHOULD be constructed using the following algorithm: \n <OrgID>: <LocalID> \nWhere <OrgID> and <LocalID> are separated by a colon ':', and where <OrgID> must include a copyrighted, trademarked or otherwise unique name that is owned by the business entity creating/ defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority.

Table 37. IBMTSSVC_StorageConfigurationCapabilities properties (continued)

Property	Type	Qualifier	Description
Supported Asynchronous Actions	Uint16[]	ModelCorrespondence (CIM_StorageConfiguration Capabilities.Supported SynchronousActions)	Enumeration indicating what operations will be executed as asynchronous jobs. If an operation is included in both this and Supported Synchronous Actions, then the underlying implementation is indicating that it may or may not create a job. **Code** **Semantics** **2** Storage Pool Creation **3** Storage Pool Deletion **4** Storage Pool Modification **5** Storage Element Creation **6** Storage Element Return **7** Storage Element Modification **8** Replica Creation **9** Replica Modification **10** Replica Attachment
SupportedCopy Types	Uint16[]	ModelCorrespondence (CIM_StorageConfiguration Service.CreateReplica. CopyType)	Describes the replication capabilities supported by the associated Storage Configuration Services. **Code** **Semantics** **2** Async **3** Sync **4** UnSyncAssoc **5** UnSyncUn Assoc **..** DMTF Reserved **0x8000..0xFFFF** Vendor Specific

Table 37. IBMTSSVC_StorageConfigurationCapabilities properties (continued)

Property	Type	Qualifier	Description
SupportedStorage ElementFeatures	uint16[]	ModelCorrespondence (CIM_StorageConfiguration Service.CreateOrModify ElementFromStoragePool. ElementType CIM_Storage Configuration Service.CreateOrModify ElementFromStoragePool. InPools)	Enumeration indicating features supported by the Storage Element methods. **Code** **Semantics** **2** StorageExtent Creation **3** StorageVolume Creation **4** StorageExtent Modification **5** StorageVolume Modification **6** Single InPool **7** Multiple InPools **..** DMTF Reserved **0x8000..0xFFFF** Vendor Specific
SupportedStorage ElementTypes	Uint16[]	ModelCorrespondence (CIM_StorageConfiguration Service.CreateOrModify ElementFromStoragePool. ElementType)	Enumeration indicating the type of storage elements that are supported by the associated Storage Configuration Service. **Code** **Semantics** **2** StorageVolume **3** StorageExtent **..** DMTF Reserved **0x8000..0xFFFF** Vendor Specific
SupportedStorage PoolFeatures	Uint16[]	ModelCorrespondence (CIM_StorageConfiguration Service.CreateOrModify StoragePool.InPools CIM_StorageConfiguration Service.CreateOrModify StoragePool.InElements)	Enumeration indicating features supported by the StoragePool methods. **Code** **Semantics** **2** InExtents **3** Single InPool **4** Multiple InPools **..** DMTF Reserved **0x8000..0xFFFF** Vendor Specific

Table 37. IBMTSSVC_StorageConfigurationCapabilities properties (continued)

Property	Type	Qualifier	Description
Supported Synchronous Actions	uint16[]	ModelCorrespondence (CIM_StorageConfiguration Capabilities.Supported AsynchronousActions)	Enumeration indicating what operations will be executed without the creation of a job. If an operation is included in both this and Supported Asynchronous Actions, then the underlying implementation is indicating that it may or may not create a job. **Code** **Semantics** **2** Storage Pool Creation **3** Storage Pool Deletion **4** Storage Pool Modification **5** Storage Element Creation **6** Storage Element Return **7** Storage Element Modification **8** Replica Creation **9** Replica Modification **10** Replica Attachment

IBMTSSVC_StorageHardwareID

The IBMTSSVC_StorageHardwareID class identifies the host port to which access to volumes can be granted.

Properties

The IBMTSSVC_StorageHardwareID class extends the CIM_StorageHardwareID class and has the properties shown in Table 38.

Table 38. IBMTSSVC_StorageHardwareID properties

Property	Type	Qualifier	Description
InstanceID	String	Expensive(TRUE)	Opaquely identifies a unique instance of Identity. The InstanceID must be unique within a namespace. In order to ensure this, the value of InstanceID should be constructed in the format: (Vendor/Admin ID):(ID) \n

Table 38. IBMTSSVC_StorageHardwareID properties (continued)

Property	Type	Qualifier	Description
Caption	String		A short textual description (one-line string) of the object
CurrentlyAuthenticated	Boolean		Indicates whether the port is currently logged in to the fibre-channel network
Description	String		A textual description of the object
ElementName	String	Expensive(TRUE)	A user-friendly name for the object
IDType	Uint16	Required(TRUE), ModelCorrespondence (CIM_StorageHardware ID.StorageID)	The type of ID property **Code** **Semantics** **1** Other **2** PortWWN **3** NodeWWN **4** Hostname
OtherIDType	String	Required(TRUE), MaxLen(256), ModelCorrespondence (CIM_StorageHardwareID. IDType), Expensive(TRUE)	A string describing the ID type if IDType is set to ″Other.″

IBMTSSVC_StoragePool

The IBMTSSVC_StoragePool class represents a group of IBMTSSVC_BackendVolume instances that aggregate to become an IBMTSSVC_StoragePool from which IBMTSSVC_StorageVolumes can be allocated.

Properties

The IBMTSSVC_StoragePool class extends the CIM_StoragePool class and has the properties shown in Table 39.

Table 39. IBMTSSVC_StoragePool properties

Property	Type	Qualifier	Description
InstanceID	String		Labels the object instance in the format cluster_id:object_id
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	A short (one-line string) textual description of the object
Description	String		A textual description of the object
ElementName	String		Name of the cluster
ExtentSize	Uint16	Units(megabytes)	The extent size of the pool. Volumes allocated from this pool have a space occupation that is a multiple of the extent size.

Table 39. IBMTSSVC_StoragePool properties (continued)

Property	Type	Qualifier	Description
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(1024)	The globally unique ID of the pool, in the format is (Vendor)(id). For the SAN Volume Controller, the ID is IBMTSSVC(id).
NativeStatus	Uint16		The native operational state of the pool **Code** **Semantics** **1** Offline **1** Online **2** Degraded
NumberOfBackend Volumes	Uint16	Counter(TRUE)	The number of BackendVolumes that make up the pool
NumberOfStorage Volumes	Uint16	Counter(TRUE)	The number of StorageVolumes allocated from the pool
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.Status Descriptions)	The pool's operational status. Values are 2 (OK) for online, 10 (Stopped) for offline, 1 (Other) for empty, or 1 (Other) for invalid. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
PoolID	String	Required(TRUE), MaxLen(256)	The ID of the pool. This ID is numeric and only unique in terms of the hosting SAN Volume Controller cluster.

Table 39. IBMTSSVC_StoragePool properties (continued)

Property	Type	Qualifier	Description
Primordial	Boolean		If true, indicates that the containing System does not have the ability to create or delete this operational element. This is important because higher level StoragePools may be assembled using the Component or AllocatedFrom StoragePool associations. Although the higher level abstractions can be created and deleted, the most basic (primordial), hardware-based StoragePools cannot. They are physically realized as part of the System or are actually managed by some other System and imported as if they were physically realized.
RemainingManaged Space	Uint64	Units(Bytes), ModelCorrespondence (StoragePool.Total ManagedSpace AllocatedFromStorage Pool.SpaceConsumed), Required(TRUE)	The remaining amount of raw storage (in bytes) from the TotalManagedSpace of this StoragePool.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.Operational Status)	Deprecated property set to ″Unknown″. Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.Operational Status)	A string describing the status that is used when the OperationalStatus property is set to 1 (Other).
TotalManaged Space	Uint64	Units(Bytes), ModelCorrespondence (StoragePool.Remaining ManagedSpace)	The total amount of raw storage (in bytes) managed by this StoragePool.

IBMTSSVC_StorageSetting

The IBMTSSVC_StorageSetting class is roughly equivalent to a service level agreement (SLA).

Properties

It defines the characteristics, qualities of service, and goals when used in a CreateOrModifyElementFromStoragePool or CreateOrModifyStoragePool method in the StorageConfigurationService. It specifies a series of properties with maximum and minimum values that define the (inclusive) bounds that the object should maintain. The setting is associated to a StorageVolume using ElementSetting.

The IBMTSSVC_StorageSetting class extends the CIM_StorageSetting class and has the properties shown in Table 40.

Table 40. IBMTSSVC_StorageSetting properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	A short textual description (one-line string) of the object
DataRedundancy Goal	Uint16	Write(TRUE), MinValue(1), ModelCorrespondence (CIM_StorageSetting.Data RedundancyMax CIM_ StorageSetting.Data RedundancyMin)	The desired number of complete copies of data to be maintained. For example, RAID 5 where one copy is maintained and RAID 1 where two or more copies are maintained. Possible values are 1 to n. The bounds for redundancy are defined using the properties DataRedundancy Max and DataRedundancy Min.
DataRedundancy Max	Uint16	Write(TRUE), MinValue(1) ,ModelCorrespondence (CIM_StorageSetting.Data RedundancyMin CIM_ StorageSetting.Data RedundancyGoal)	The maximum number of complete copies of data to be maintained. For example, RAID 5 where one copy is maintained and RAID 1 where two or more copies are maintained. Possible values are 1 to n.
DataRedundancy Min	Uint16	Write(TRUE), MinValue(1), ModelCorrespondence (CIM_StorageSetting.Data RedundancyMax CIM_StorageSetting.Data RedundancyGoal)	The minimum number of complete copies of data to be maintained. For example, RAID 5 where one copy is maintained and RAID 1 where two or more copies are maintained. Possible values are 1 to n.

Table 40. IBMTSSVC_StorageSetting properties (continued)

Property	Type	Qualifier	Description
DeltaReservation Goal	Uint8	Units(Percentage), MinValue(1), MaxValue(100), ModelCorrespondence (CIM_StorageSetting.Delta ReservationMinCIM_ StorageSetting.Delta ReservationMax), Write(TRUE)	The amount of space that should be reserved in a replica for caching changes, specifed by a number between 1 (1%) and 100 (100%). For a complete copy, use 100. The bounds for the reservation are defined using the properties DeltaReservation Max and DeltaReservation Min.
DeltaReservation Max	Uint8	Units(Percentage), MinValue(1), MaxValue(100), ModelCorrespondence (CIM_StorageSetting.Delta ReservationMinCIM_ StorageSetting.Delta ReservationGoal), Write(TRUE)	The maximum amount of space that should be reserved in a replica for caching changes, specifed by a number between 1 (1%) and 100 (100%). For a complete copy, use 100.
DeltaReservation Min	Uint8	Units(Percentage), MinValue(1), MaxValue(100), ModelCorrespondence (CIM_StorageSetting.Delta ReservationMaxCIM_ StorageSetting.Delta ReservationGoal), Write(TRUE)	The minimum amount of space that should be reserved in a replica for caching changes, specifed by a number between 1 (1%) and 100 (100%). For a complete copy, use 100.
Description	String		A textual description of the object
ElementName	String	Required(TRUE)	The user-friendly name for this instance of SettingData. In addition, the user-friendly name can be used as an index property for a search or query. The name does not have to be unique within a namespace.

Table 40. IBMTSSVC_StorageSetting properties (continued)

Property	Type	Qualifier	Description
InstanceID	String		Within the scope of the instantiating Namespace, InstanceID uniquely identifies an instance. The ID should be constructed using the following algorithm: <OrgID>:<LocalID>. <OrgID> must include a copyrighted, trademarked, or otherwise unique name that is owned by the business entity that is creating or defining the ID or it must be a registered ID that is assigned by a global authority. <OrgID> must not contain a colon (:). <LocalID> is chosen by the business entity and should not be re-used to identify different underlying (real-world) elements.
NoSinglePoint OfFailure	Boolean	Write(TRUE)	The desired value for No Single Point of Failure. Possible values are false (single point of failure) and true (no single point of failure).
PackageRedundancy Goal	Uint16	Write(TRUE), ModelCorrespondence (CIM_StorageSetting. PackageRedundancyMax CIM_StorageSetting. PackageRedundancyMin)	The desired number of redundant packages to use. Possible values are 0 to n. For example, in the storage domain package redundancy describes the number of disk spindles that can fail without data loss including, at most, one spare. For example, a RAID 5 with a spare disk could have a PackageRedundancy of 2.

Table 40. IBMTSSVC_StorageSetting properties (continued)

Property	Type	Qualifier	Description
PackageRedundancy Max	Uint16	Write(TRUE), ModelCorrespondence (CIM_StorageSetting. PackageRedundancyMin CIM_StorageSetting. PackageRedundancyGoal)	PackageRedundancy Max describes the maximum number of redundant packages to use. Possible values are 0 to n.
PackageRedundancy Min	Uint16	Write(TRUE), ModelCorrespondence (CIM_StorageSetting. PackageRedundancyMax CIM_StorageSetting. PackageRedundancyGoal)	The minimum number of redundant packages to use. Possible values are 0 to n.

IBMTSSVC_StorageVolume

The IBMTSSVC_StorageVolume class represents a device presented by the Cluster that can be mapped as a SCSI LUN to host systems on the SAN. A Volume is formed by allocating a set of Extents from a Pool.

Properties

The IBMTSSVC_StorageVolume class extends the CIM_StorageVolume class and has the properties shown in Table 41.

Table 41. IBMTSSVC_StorageVolume properties

Property	Type	Qualifier	Description
Access	Uint16		Describes the media **Code** **Semantics** **0** Unknown **1** Readable **2** Writeable **3** Read/Write Supported **4** Write Once
AccessGranted	Boolean	ModelCorrespondence (CIM_Controller. AuthorizationView)	A quick interface for finding Devices with no AuthorizationSubject association to an AccessControl Information instance, either directly or via a Controller. True indicates that the Device has granted access to a consumer. False indicates that no access has been granted.

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
Additional Availability	Uint16[]	Deprecated(CIM_Associated PowerManagementService. PowerState CIM_Managed SystemElement.Operational Status CIM_EnabledLogical Element.EnabledState), ModelCorrespondence (CIM_LogicalDevice. Availability)	Additional availability and status of the Device, beyond that specified in the Availability property. The Availability property denotes the primary status and availability of the Device. In some cases, it is not sufficient to denote the complete status of the Device. In those cases, the AdditionalAvailability property can be used to provide further information. **Code** **Semantics** **1** Other **2** Unknown **3** Running/Full Power **4** Warning **5** In Test **6** Not Applicable **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
Availability	Uint16	Deprecated(CIM_Associated PowerManagement Service.PowerState CIM_ManagedSystem Element.OperationalStatus CIM_EnabledLogical Element.EnabledStatus), ModelCorrespondence (CIM_LogicalDevice. AdditionalAvailability)	The primary availability and status of the Device **Code** **Semantics** **1** Other **2** Unknown **3** Running/Full Power **4** Warning **5** In Test **6** Not Applicable **7** Power Off **8** Off Line **9** Off Duty **10** Degraded **11** Not Installed **12** Install Error **13** Power Save - Unknown **14** Power Save - Low Power Mode **15** Power Save - Standby **16** Power Cycle **17** Power Save - Warning **18** Paused **19** Not Ready **20** Not Configured **21** Quiesced
BackendVolumeID	String	Expensive(TRUE)	The ID of the underlying BackendVolume. Only valid if Type=Image.
BackendVolume Name	String	Expensive(TRUE)	The name of the underlying BackendVolume. Only valid if Type=Image.
BlockSize	Uint64	Units(Bytes)	Size in bytes of the blocks that form this StorageExtent. If variable block size, then the maximum block size in bytes should be specified. If the block size is unknown or if a block concept is not valid (for example, for AggregateExtents, Memory, or LogicalDisks), enter a 1.
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	A short textual description (one-line string) of the object

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
ConsumableBlocks	Uint64		The maximum number of blocks, of size BlockSize, that are available for consumption when layering StorageExtents using the BasedOn association. This property only has meaning when this StorageExtent is an Antecedent reference in a BasedOn relationship. For example, a StorageExtent could be composed of 120 blocks. However, the Extent itself may use 20 blocks for redundancy data. If another StorageExtent is BasedOn this Extent, only 100 blocks would be available to it. This information (″100 blocks are available for consumption″) is indicated in the ConsumableBlocks property.
Controlled	Boolean		A quick interface for finding Devices with no ControlledBy associations to Controllers. True indicates that the Device is connected to one or more Ports (via Controllers). False indicates that the Device exists but is not connected to a port.
CreationClass Name	String	MaxLen(256)	The name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
DataOrganization	Uint16		Type of data organization used **Code** **Semantics** **0** Other **1** Unknown **2** Fixed Block **3** Variable Block **4** Count Key Data
DataRedundancy	Uint16	ModelCorrespondence (CIM_StorageSetting.Data RedundancyGoal CIM_StorageSetting. DataRedundancyMax CIM_StorageSetting.Data RedundancyMin)	The number of complete copies of data that are maintained
DeltaReservation	Uint8	MinValue(0), MaxValue(100), Units(Percentage), ModelCorrespondence (CIM_StorageSetting.Delta ReservationGoal CIM_StorageSetting.Delta ReservationMax CIM_StorageSetting.Delta ReservationMin)	The current value for Delta reservation
Description	String		A textual description of the object
DeviceID	String	MaxLen(64)	The ID of the StorageVolume. A numeric value that is unique for intances of the StorageVolume class only.
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	The user-friendly name of the volume
EnabledDefault	Uint16	Write(TRUE)	The default or startup configuration for an element's EnabledStatus. By default, the EnabledStatus is 2 (Enabled). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	The operational state of the element **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved If an element is being tested and is neither enabled or disabled, then In Test (7) is used. If this property does not apply to an instance of EnabledLogical Element, then Not Applicable (5) is used.
ErrorCleared	Boolean	Deprecated(CIM_Managed SystemElement.Operational Status)	Unsupported property
ErrorDescription	String	Deprecated(CIM_Device ErrorData.ErrorDescription)	Unsupported property
ErrorMethodology	String		Unsupported property

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
ExtentStatus	Uint16[]		StorageExtents have additional status information beyond that captured in the Availability and StatusInfo properties, inherited from the ManagedSystem Element. This additional information (for example, ″Protection Disabled,″ value=9) is captured in the VolumeStatus property. **Code** **Semantics** **0** Other **1** Unknown **2** None/Not Applicable **3** Broken **4** Data Lost **5** Dynamic Reconfig **6** Exposed **7** Fractionally Exposed **8** Partially Exposed **9** Protection Disabled **10** Readying **11** Rebuild **12** Recalculate **13** Spare in Use **14** Verify In Progress **15..32767** DMTF Reserved **32768..65535** Vendor Reserved
FCID	String		The Flash Copy ID of the volume
FCName	String		The Flash Copy name of the volume
GroupID	String		The ID of the scoping RedundancyGroup
GroupName	String		The name of the scoping RedundancyGroup

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
Identifying Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice.Other IdentifyingInfo)	An array of free-form Strings that provide explanations and details behind the entries in the OtherIdentifyingInfo array. Each entry of this array is related to the entry in OtherIdentifyingInfo that is located at the same index.
InstallDate	Date-time		Unsupported property
IsBasedOn Underlying Redundancy	Boolean		True indicates that the underlying StorageExtent(s) participate in a StorageRedundancy Group
IsFormatted	Boolean	Expensive(TRUE)	True indicates that the volume has been formatted by SAN Volume Controller
LastErrorCode	Uint32	Deprecated(CIM_Device ErrorData.LastErrorCode)	Unsupported property
MaxQuiesceTime	Uint64	Deprecated(No value), Units(MilliSeconds)	Unsupported property
Name	String	ModelCorrespondence (CIM_StorageVolume. NameFormat), MaxLen(1024)	A unique identifier for the Volume

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
NameFormat	Uint16	ModelCorrespondence (CIM_StorageVolume.Name CIM_StorageVolume. OtherNameFormat), Experimental(TRUE)	Format of the Name property. For non SCSI volumes, SNVM may be the most appropriate choice. **Code** **Semantics** **0** Unknown **1** Other **2** VPD83NAA6 (VPD Page 83, NAA IEEE Registered Extended) **3** VPD83NAA5 (VPD Page 83, NAA IEEE Registered) **4** VPD83Type2 (VPD Page 83, EIU-64) **5** VPD83Type1 (VPD Page 83, T10 Vendor Identification) **6** VPD83Type0 **7** SNVM (Serial Number/ Vendor/ Model. VPD Page 83, Vendor-Specific) **8** NodeWWN (Node WWN, for single LUN or controller)

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
NameNamespace	Uint16	ModelCorrespondence (CIM_StorageVolume.Name CIM_StorageVolume. OtherNameNamespace)	The preferred source for volume names is SCSI VPD Page 83 responses. Page 83 returns a list of identifiers for various device elements. The metadata for each identifier includes an Association field, identifiers with association of 0 apply to volumes. Page 83 supports several namespaces specified in the Type field in the identifier metadata. **Code** **Semantics** **0** Unknown **1** Other **2** VPD83Type3n (Page 83, Type 3 NAA. NameFormat should be NAA.) **3** VPD83Type2 (VPD Page 83, Type 2 EUI64. NameFormat EUI.) **4** VPD83Type1 (VPD Page 83, Type 1 T10 Vendor Identification. NameFormat T10.) **5** VPD80 (VPD page 80, Serial number. NameFormat should be Other.) **6** NodeWWN (FC NodeWWN. NameFormat should be NAA or EUI.) **7** SNVM (Serial Number/ Vendor/ Model. NameFormat should be SNVM.)

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
NativeStatus	Uint16		The native operational status of the volume **Code** **Semantics** **0** Offline **1** Online **2** Degraded
NoSinglePoint OfFailure	Boolean	ModelCorrespondence (CIM_StorageSetting.No SinglePointOfFailure)	Indicates whether or not a single point of failure exists
NumberOfBlocks	Uint64		Total number of logically contiguous blocks, of size BlockSize, that form this Extent. The total size of the Extent can be calculated by multiplying BlockSize by NumberOfBlocks. If the BlockSize is 1, this property is the total size of the Extent.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The status of the volume **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	The state of the element when the EnabledStatus property is set to 1 (Other). This property must be set to null when EnabledStatus is any value other than 1.

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
OtherIdentifyinInfo	String[]	MaxLen(256), ArrayType(Indexed), ModelCorrespondence (CIM_LogicalDevice. IdentifyingDescriptions)	Additional data, beyond DeviceID information, that can be used to identify a LogicalDevice. For example, the Operating System user friendly name for the Device.
OtherNameFormat	String	ModelCorrespondence (CIM_StorageVolume. NameFormat)	The description of the format of the Name property when NameFormat includes the value 1 (Other)
OtherName Namespace	String	ModelCorrespondence (CIM_StorageVolume. NameNamespace)	The description of the namespace of the Name property when NameNamespace includes the value 1 (Other)
PackageRedundancy	Uint16	ModelCorrespondence (CIM_StorageSetting. PackageRedundancyGoal CIM_StorageSetting. PackageRedundancyMax CIM_StorageSetting. PackageRedundancyMin)	The number of disk spindles that can fail without data loss
PoolID	String		The ID of the hosting storage pool
PoolName	String		The name of the pool from which this volume was allocated

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
PowerManagement Capabilities	Uint16[]	Deprecated(CIM_Power ManagementCapabilities. PowerCapabilities)	An enumerated array describing the power management capabilities of the Device. This property has been deprecated. Instead, the PowerCapabilites property in an associated PowerManagement Capabilities class should be used. **Code** **Semantics** **0** Unknown **1** Not Supported **2** Disabled **3** Enabled **4** Power Saving Modes Entered Automatically **5** Power State Settable **6** Power Cycling Supported **7** Timed Power On Supported
PowerManagement Supported	Boolean	Deprecated(CIM_Power ManagementCapabilities)	Boolean indicating that the Device can be power managed. This property has been deprecated. Instead, the existence of an associated PowerManagement Capabilities class (associated using the ElementCapabilities relationhip) indicates that power management is supported.
PowerOnHours	Uint64	Deprecated(CIM_Powered StatisticalData.PowerOnHours), Units(Hours), Counter(TRUE)	Unsupported property
PreferredNode	String	Expensive(TRUE)	The ID of the preferred node.
Primordial	Boolean		Indicates whether the containing System does not have the ability to create or delete this operational element
Purpose	String		The description of the media and its use

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Indicates the state change for the element that should be made at the next opportunity. The SAN Volume Controller does not evaluate this attribute and therefore no action is taken when it changes. **Code Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
SCID	String		The sync copy ID of the volume
SCName	String		The sync copy name of the volume
SequentialAccess	Boolean		Indicates whether the Storage is sequentially accessed by a MediaAccessDevice. For example, a TapePartition is a sequentially accessed StorageExtent. StorageVolumes, DiskPartitions, and LogicalDisks are random-access Extents.
Status	String	MaxLen(10), Deprecated(CIM_ ManagedSystemElement. OperationalStatus)	Deprecated property that is set to Unknown. See OperationalStatus for status information
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	The description of the status that is used when the OperationalStatus property is set to 1 (Other)

Table 41. IBMTSSVC_StorageVolume properties (continued)

Property	Type	Qualifier	Description
StatusInfo	Uint16	Deprecated(CIM_Enabled LogicalElement.Enabled State)	Deprecated porperty. See CIM_Enabled LogicalElement. EnabledState instead. **Code** **Semantics** **1** Other **2** Unknown **3** Enabled **4** Disabled **5** Not Applicable
SystemCreation ClassName	String	Propagated(CIM_System. CreationClassName), MaxLen(256)	The CreationClassName of the scoping System
SystemName	String	Propagated(CIM_System. Name), MaxLen(256)	The IP address of the scoping cluster
Throttle	Uint64	Units(IOs per second), Write(TRUE), WriteRole(Administrator), Expensive(TRUE)	The maximum bandwidth of the volume
TimeOfLast StateChange	Date-time		Unsupported property
TotalPowerOn Hours	Uint64	Deprecated(CIM_Powered StatisticalData.TotalPower OnHours), Units(Hours), Counter(TRUE)	Unsupported Property
Type	Uint32	Value(Sequential Striped Router Image)	The type of the volume

IBMTSSVC_SyncCopySynchronizedSet

The IBMTSSVC_SyncCopySynchronizedSet class aggregates multiple StorageSynchronized instances to ensure consistent copying.

Properties

The IBMTSSVC_SyncCopySynchronizedSet class extends the CIM_SynchronizedSet class and has the properties shown in Table 42.

Table 42. IBMTSSVC_SyncCopySynchronizedSet properties

Property	Type	Qualifier	Description
AuxilaryID	String		The ID of the auxilary cluster
AuxilaryName	String		The name of the auxilary cluster
Availability	Uint32	Expensive(TRUE)	The availability of the set **Code** **Semantics** **0** Online **1** Primary Offline **2** Secondary Offline **3** IO Channel Offline

Table 42. IBMTSSVC_SyncCopySynchronizedSet properties (continued)

Property	Type	Qualifier	Description
Caption	String	MaxLen(15), Write(TRUE), WriteRole(Administrator)	A short (one-line string) textual description
Connected	Boolean		The status of the network connection
CopyType	Uint16	Experimental(TRUE)	The replication policy of the SynchronizedSet **Code** **Semantics** **2** Async. Create and maintain an asynchronous copy of the source. **3** Sync. Create and maintain a synchronized copy of the source. **4** UnSyncAssoc **..** DMTF Reserved **0x8000..** Vendor Specific
Description	String		A textual description of the object
ElementCount	Uint32	Counter	A the number of SyncCopyStorage Synchronized in this set
ElementName	String	MaxLen(15), Write(TRUE), WriteRole(Administrator), Experimental(TRUE)	The user-friendly name for this instance of SynchronizedSet. In addition, the user-friendly name can be used as a property for a search or query. ElementName does not have to be unique within a namespace.
FreezeTime	String	Expensive(TRUE)	The time the relationship was stopped
InstanceID	String		InstanceID opaquely identifies a unique instance of collection that is scoped (contained) by a System. The InstanceID must be unique within a namespace. In order to ensure uniqueness, the value of InstanceID should be constructed in the following format: \n(Vendor ID)(ID) \n
MasterID	String		The ID of the master cluster

Table 42. IBMTSSVC_SyncCopySynchronizedSet properties (continued)

Property	Type	Qualifier	Description
NativeState	Uint16	ValueMap, Values	The native state of the set **Code** **Semantics** **0** Idling **1** Idling disconnected **2** Consistent synchronized **3** Consistent disconnected **4** Consistent stopped **5** Inconsistent copying **6** Inconsistent disconnected **7** Inconsistent disconnected **8** Empty
Primary	Uint32		Shows which side is currently the primary in the relationship. The primary volumes are the ones accessible for I/O by the clients.
Status	Uint32		The status of the SynchronizedSet **Code** **Semantics** **4** Prepared **5** ReSyncIn Progress **6** Synchronized **12** Broken **13** Fractured **0x1000** Empty **0x8101** Fractured Idle
SyncMaintained	Boolean	Expensive(TRUE)	Indicates whether the synchronization relationship is maintained

Service object classes

The service classes and their properties of the CIM Agent for the SAN Volume Controller are described in the following pages.

IBMTSSVC_ClusteringService

The IBMTSSVC_ClusteringService class provides the methods for managing the SAN Volume Controller cluster, such as adding or removing nodes.

Properties

The IBMTSSVC_ClusteringService class extends the CIM_ClusteringService class and has the properties shown in Table 43 on page 192.

Table 43. IBMTSSVC_ClusteringService properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Unsupported property
CreationClassName	String	MaxLen(256)	Indicates the name of the class or subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		Provides a textual description of the object.
ElementName	String		Unsupported property
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled.″ **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 43. IBMTSSVC_ClusteringService properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicating one of the following: **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(256)	The label by which the object is known.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The operational status of the service. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode

Table 43. IBMTSSVC_ClusteringService properties (continued)

Property	Type	Qualifier	Description
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledState property is set to 1 (″Other″). This property MUST be set to NULL when EnabledState is any value other than 1.
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	RequestedStatus is an integer enumeration indicating whether the element should be shut down, enabled, disabled, taken offline, or tested at the next opportunity. **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Started	Boolean		Indicates if this service is started.
StartMode	String	MaxLen(10), Deprecated (CIM_Service.EnabledDefault)	Indicates if this service is started manually or automatic.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	Describes the status - used when the OperationalStatus property is set to ″Other.″
SystemCreation ClassName	String	Propagated(CIM_System. CreationClassName), MaxLen(256)	The scoping system's creation class name.

Table 43. IBMTSSVC_ClusteringService properties (continued)

Property	Type	Qualifier	Description
SystemName	String	Propagated (CIM_System.Name), MaxLen(256)	The scoping system's name.
TimeOfLastState Change	Date-time		Unsupported property

IBMTSSVC_PrivilegeManagementService

IBMTSSVC_PrivilegeManagementService class extends the CIM_PrivilegeManagementService class.

Properties

The IBMTSSVC_PrivilegeManagementService class has the properties shown in Table 44.

Table 44. IBMTSSVC_PrivilegeManagementService properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Unsupported property
CreationClassName	String	MaxLen(256)	Indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		Provides a textual description of the object.
ElementName	String		Unsupported property

Table 44. IBMTSSVC_PrivilegeManagementService properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(256)	The label by which the object is known.

Table 44. IBMTSSVC_PrivilegeManagementService properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The operational status of the service. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabled State	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledStatus property is set to 1 (″Other″). This property must be set to NULL when EnabledStatus is any value other than 1.
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property

Table 44. IBMTSSVC_PrivilegeManagementService properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. This property is provided to compare Requested and current Enabled statuses. Note that when EnabledStatus is set to 5 (″Not Applicable″), writing this property has no effect. The default is 5 (″No Change″). **Code Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Started	Boolean		Indicates if this service is started.
StartMode	String	MaxLen(10), Deprecated (CIM_Service.Enabled Default)	Indicates if this service is started manually or automatic.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status - used when the OperationalStatus property is set to 1 (″Other″).
SystemCreation ClassName	String	Propagated (CIM_System.Creation ClassName), MaxLen(256)	The scoping system's creation class name.
SystemName	String	Propagated (CIM_System.Name), MaxLen(256)	The scoping system's name.
TimeOfLast StateChange	Date-time		Unsupported property

IBMTSSVC_StorageConfigurationService

IBMTSSVC_StorageConfigurationService class provides extrinsic methods for basic storage configuration tasks.

Properties

The IBMTSSVC_StorageConfigurationService class extends the CIM_StorageConfigurationService class and has the properties shown in Table 45.

Table 45. IBMTSSVC_StorageConfigurationService properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Unsupported property
CreationClassName	String	MaxLen(256)	Indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		Provides a textual description of the object.
ElementName	String		Unsupported property
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating an administrator's default/ startup configuration for an element's EnabledStatus. By default, the element is ″Enabled″ (value=2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 45. IBMTSSVC_StorageConfigurationService properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	Integer enumeration indicator. **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(256)	The label by which the object is known.
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The operational status of the service. **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode

Table 45. IBMTSSVC_StorageConfigurationService properties (continued)

Property	Type	Qualifier	Description
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	A string describing the element's enabled/disabled state when the EnabledStatus property is set to 1 (″Other″). This property must be set to NULL when EnabledStatus is any value other than 1.
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Integer enumeration indicator. **Code Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Started	Boolean		Indicates if this service is started.
StartMode	String	MaxLen(10), Deprecated (CIM_Service.Enabled Default)	Indicates if this service is started manually or automatic.
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem Element.OperationalStatus)	Deprecated property - set to ″Unknown.″ Look at OperationalStatus for status information
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	A string describing the status. Is used when the OperationalStatus property is set to 1 (″Other″).
SystemCreation ClassName	String	Propagated (CIM_System.Creation ClassName), MaxLen(256)	The scoping system's creation class name.
SystemName	String	Propagated (CIM_System.Name), MaxLen(256)	The scoping system's name.
TimeOfLast StateChange	Date-time		Unsupported property

IBMTSSVC_StorageHardwareIDManagementService

This service provides extrinsic methods to manage HardwareAccounts and Hosts for the SAN Volume Controller.

Properties

The IBMTSSVC_StorageHardwareIDManagementService class extends the CIM_StorageHardwareIDManagementService class and has the properties shown in Table 46.

Table 46. IBMTSSVC_StorageHardwareIDManagementService properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Unsupported property
CreationClass Name	String	MaxLen(256)	The name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.
Description	String		A textual description of the object
ElementName	String		Unsupported property
EnabledDefault	Uint16	Write(TRUE)	An enumerated value indicating the default or startup EnabledStatus. By default, the element is Enabled (2). **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768..65535** Vendor Reserved

Table 46. IBMTSSVC_StorageHardwareIDManagementService properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.OtherEnabledState)	The operational state of the element **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768..65535** Vendor Reserved If an element is being tested and is neither enabled or disabled, then In Test (7) is used. If this property does not apply to an instance of EnabledLogical Element, then Not Applicable (5) is used.
InstallDate	Date-time		Unsupported property
Name	String	MaxLen(256)	The name of the object

Table 46. IBMTSSVC_StorageHardwareIDManagementService properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.StatusDescriptions)	The operational status of the service **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Completed **18** Power Mode
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	The state of the element when the EnabledStatus property is set to 1 (Other). This property must be set to null when EnabledStatus is any value other than 1.
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Unsupported property
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Unsupported property

Table 46. IBMTSSVC_StorageHardwareIDManagementService properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLogical Element.EnabledState)	Indicates the state change for the element that should be made at the next opportunity. When EnabledStatus is set to 5 (No Change), then this property has no effect. By default, the RequestedStatus is 5 (No Change). **Code Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10..32767** DMTF Reserved **32768..65535** Vendor Reserved
Started	Boolean		Indicates if the service is started
StartMode	String	MaxLen(10), Deprecated (CIM_Service.EnabledDefault)	Indicates if the service is started manually or automatically
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystemElement. OperationalStatus)	This property is deprecated and set to ″Unknown.″ See OperationalStatus for status information.
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem Element.OperationalStatus)	The description of the status that is used when the OperationalStatus property is set to 1 (Other)
SystemCreation ClassName	String	Propagated(CIM_System. CreationClassName), MaxLen(256)	The scoping system's creation class name
SystemName	String	Propagated(CIM_System. Name), MaxLen(256)	The name of the scoping system
TimeOfLast StateChange	Date-time		Unsupported property

Security object classes

This section describes the security classes and their properties of the CIM Agent for the SAN Volume Controller.

This section describes the security classes and their properties of the CIM Agent for the SAN Volume Controller.

IBMTS_Account

The IBMTS_Account class extends the CIM_Account class.

Properties

The IBMTS_Account class represents a single user account on the Common Information Model Object Model (CIMOM) and stores the authentication (user name and password) and authorization (global and system roles) information.

The IBMTS_Account class has the properties shown in Table 47.

Table 47. IBMTS_Account properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
CreationClass Name	String	MaxLen(256), ReadRole(None)	Indicates the name of the class or the subclass used in creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified
Description	String		Provides a textual description of the object
Descriptions	String[]	MaxLen(1024)	Contains descriptions of the object that the user can read. In the case of an LDAP-derived instance, the description attribute may have multiple values that, therefore, cannot be placed in the inherited Description property
ElementName	String		Defines the name of the object. This property enables each instance to define a user-friendly name in addition to its key properties or identity data, and description information
Host	String[]		Specifies the name(s) of the system(s) to which the account applies. The host name may be a fully-qualified DNS name or it may be an unqualified host name

Table 47. IBMTS_Account properties (continued)

Property	Type	Qualifier	Description
InstallDate	Date-time		Specifies the date when the object was installed. Lack of this value does not mean that the object was not installed
LocalityName	String[]		Specifies the name of the selected locality, such as the name of a city, county or other geographic region
Name	String	MaxLen(1024) ReadRole(None)	Specifies the name of the object instance. The value of this property may be set to be the same as that of the UserID property or, in the case of an LDAP-derived instance, the Name property value may be set to the DistinguishedName of the LDAP-accessed object instance.
ObjectClass	String[]		In the case of an LDAP-derived instance, this property value(s) may be set to the objectClass attribute values
Operational Status	Uint16[]	Experimental(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Indicates the current status of the element **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity Error **17** Complete **18** Power Mode

Table 47. IBMTS_Account properties (continued)

Property	Type	Qualifier	Description
Organization Name	String[]	Required(TRUE)	Specifies the name of the organization related to the account
OU	String[]		Specifies the name of an organizational unit related to the account
Status Descriptions	String[]	Experimental(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Describes the status of the element when OperationalStatus is set to 1
SeeAlso	String[]		Specifies the distinguished Name of other Directory objects that might resemble the real-world objects
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem)	Indicates the current status of the object
SystemCreation ClassName	String	Propagated (CIM_System. CreationsClassName), MaxLen(256), ReadRole(None)	Specifies the cluster configuration node (CCN) of the System
SystemName	String	Propagated (CIM_System.Name), MaxLen(256), ReadRole(None)	Specifies the name of the System
UserCertificate	String[]	Octetstring(TRUE)	Specifies a public key certificate for the user
UserID	String	MaxLen(256)	Defines the identifier for a user to the System
UserPassword	String[]	Octetstring(TRUE)	Contains an encrypted password for an authorized user to access the resources in a specified directory

IBMTS_AccountManagementService

The IBMTS_AccountManagementService class provides the methods for managing the accounts on the Common Information Model Object Manager (CIMOM).

Properties

The IBMTS_AccountManagementService class extends the CIM_AccountManagementService class and has the properties shown in Table 48.

Table 48. IBMTS_AccountManagementService properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64), ReadRole(None)	Identifies the object with a short (one-line string) textual description

Table 48. IBMTS_AccountManagementService properties (continued)

Property	Type	Qualifier	Description
CreationClassName	String	MaxLen(256), ReadRole(None)	Indicates the name of the class or subclass used to create an instance
SystemCreationClass	String	Propogated (CIM_System.Creation) MaxLen(256), ReadRole(None)	Indicates the system's creation class name
SystemName	String	Propogated (CIM_System.Name) MaxLen(256), ReadRole(None)	Indicates the system's name
Description	String	ReadRole(None)	Provides a textual description of the object
ElementName	String	ReadRole(None)	Specifies the name of the instance
EnabledDefault	Uint16	Write(TRUE)	Indicates the administrator's default or startup configuration for an element's Enabled Status. By default, the element is Enabled (value = 2) **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8...32767** DMTF Reserved **32768...** Vendor Reserved

Table 48. IBMTS_AccountManagementService properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLog)	Indicates whether the element is currently shutting down or in an enabled or disabled state **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11...32767** DMTF Reserved **32768...** Vendor Reserved
InstallDate	Date-time		Indicates the date on which the CIM client created the object in the CIMOM's repository
Name	String	MaxLen(256), ReadRole(None)	Defines the unique label, in the context of the hosting system, by which the AccessControl Information is known

Table 48. IBMTS_AccountManagementService properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	Experimental(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Indicates the operational status of the cluster **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Complete **18** Power Mode
OtherEnabledStatus	String	ModelCorrespondence (CIM_EnabledLog)	Describes the element's enabled or disabled state when EnabledStatus is set to 1; must be null when EnabledStatus is set to a value other than 1
StatusDescriptions	String[]	Experimental(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Describes the status of the cluster when OperationalStatus is set to 1
PrimaryOwnerContact	String	Experimental(TRUE), MaxLen(256), Write(TRUE)	Specifies how the primary owner of the account can be contacted, such as phone number or e-mail
PrimaryOwnerName	String	Experimental(TRUE), MaxLen(64), Write(TRUE)	Specifies the name of the primary owner

Table 48. IBMTS_AccountManagementService properties (continued)

Property	Type	Qualifier	Description
RequestedStatus	Uint16	Write(TRUE), ModelCorrespondence (CIM_EnabledLog)	Sets the state (shut down, enabled, taken offline, or tested) of the element at the next operation **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10...32767** DMTF Reserved **32768...** Vendor Reserved
Started	Boolean		Indicates that the Service is started or stopped for the account
StartMode	String	MaxLen(10), Deprecated (CIM_Service.Enable)	Indicates whether the Service is automatically started by a system, operating system, or specific user request
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem)	Indicates the current status of the Service instance
TimeOfLastStateChange	Date-time	Experimental(TRUE)	Indicates the time the last change of state occurred

IBMTS_CIMXMLCommunicationMechanism

The IBMTS_CIMXMLCommunicationMechanism adds properties that are specific to the CIM-XML protocol.

Properties

The IBMTS_CIMXMLCommunicationMechanism properties are shown in Table 49

Table 49. IBMTS_CIMXMLCommunicationMechanism properties

Property	Type	Qualifier	Description
Authentication Mechanism Descriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ObjectManager)	Provides a description of the supported mechanisms. An entry in this descriptions array must be provided when 1 = Other is specified

Table 49. IBMTS_CIMXMLCommunicationMechanism properties (continued)

Property	Type	Qualifier	Description
Authentication Mechanisms Supported	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ObjectManager), Required(TRUE)	Describes the type of authentication supported by the ObjectManager **Code** **Semantics** **0** Unknown **1** Other **2** None **3** Basic **4** Digest
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
CIMValidated	Boolean	Required(TRUE)	Describes whether the CIM server is strictly validating
CIMXMLProtocol Version	Uint16	Deprecated (CIM_CIMXML), Required(TRUE)	Describes the CIM-XML protocol version supported by the ObjectManager **Code** **Semantics** **0** Unknown **1** 1.0
Communication Mechanism	Uint16	Required(TRUE), ModelCorrespondence (CIM_ObjectManager)	Describes the encoding and protocol which can be used to communicate with the ObjectManager **Code** **Semantics** **0** Unknown **1** Other **2** CIM-XML
CreationClassName	String	MaxLen(256)	Indicates the name of the class or subclass used to create an instance
Description	String		Provides a textual description of the object
ElementName	String	ReadRole(None)	Specifies the name of the instance

Table 49. IBMTS_CIMXMLCommunicationMechanism properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	Indicates the administrator's default or startup configuration for an element's Enabled Status. By default, the element is Enabled (value = 2) **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8...32767** DMTF Reserved **32768...** Vendor Reserved
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLog)	Indicates whether the element is currently shutting down or in an enabled or disabled state **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11...32767** DMTF Reserved **32768...** Vendor Reserved
FunctionalProfile Description	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ObjectManager)	Provides a description of the supported operations of the ObjectManager

Table 49. IBMTS_CIMXMLCommunicationMechanism properties (continued)

Property	Type	Qualifier	Description
FunctionalProfile Supply	Uint16[]	Required(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_ObjectManager)	Enumerated array describing the types of operations supported by the ObjectManager **Code** **Semantics** **0** Unknown **1** Other **2** Basic Read **3** Basic Write **4** Schema Manipulating **5** Instance Manipulating **6** Associated Traversal **7** Query Execution **8** Qualifier Declaration **9** Indication
InstallDate	Date-time		Indicates the date on which the CIM client created the object in the CIMOM's repository
MultipleOperation Supply	Boolean	Required(TRUE)	Indicates whether the ObjectManager supports multiple operation requests (TRUE) or only simple requests (FALSE)
Name	String	MaxLen(256)	Defines the unique label, in the context of the hosting system, by which the AccessControl Information is known

Table 49. IBMTS_CIMXMLCommunicationMechanism properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Indicates the operational status of the cluster **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Complete **18** Power Mode **..** DMTF Reserved **0x8000** Vendor Reserved
OtherCommunication Mechanism	String	ModelCorrespondence (CIM_ObjectManager)	Provides a description of the supported protocols when 1 = Other, is specified in the Communication Mechansim.
OtherEnabledState	String	ModelCorrespondence (CIM_EnabledLog)	Describes the element's enabled or disabled state when EnabledStatus is set to 1; must be null when EnabledStatus is set to a value other than 1

Table 49. IBMTS_CIMXMLCommunicationMechanism properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLog)	Sets the state (shut down, enabled, taken offline, or tested) of the element at the next operation **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10** Reboot **11** Reset **..** DMTF Reserved **32768...** Vendor Reserved
Status	String	Deprecated (CIM_ManagedSystem), MaxLen(10)	Indicates the current status of the Service instance
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Describes the various OperationalStatus array values
SystemCreation Class	String	Propogated (CIM_System.Creation) MaxLen(256)	Indicates the system's creation class name
SystemName	String	Propogated (CIM_System.Name) MaxLen(256)	Indicates the system's name
TimeOfLast StateChange	Date-time		Indicates the time the last change of state occurred
Version	String	Required(TRUE)	Describes the CIM-XML protocol version supported by the ObjectManager

IBMTS_IndicationFilter

The IBMTS_IndicationFilter extends the CIM_IndicationFilter class.

Properties

The IBMTS_IndicationFilter has the properties shown in Table 50.

Table 50. IBMTS_IndicationFilter properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
CreationClassName	String	MaxLen(256)	Indicates the name of the class or subclass used to create an instance
SystemCreationClass	String	MaxLen(256)	Indicates the system's creation class name
SystemName	String	MaxLen(256)	Indicates the system's name
Description	String		Provides a textual description of the object
ElementName	String		Specifies the name of the instance
Query	String	Required(TRUE), ModelCorrespondence (CIM_IndicationFilter)	A query expression that defines the condition(s) under which indications will be generated
QueryLanguage	String	Required(TRUE)	The language in which the query is expressed
SourceNamespace	String		The path to a local namespace where the indications originate
Name	String	MaxLen(256)	Defines the unique label, in the context of the hosting system, by which the AccessControl Information is known

IBMTS_NameSpace

The IBMTS_NameSpace extends the CIM_NameSpace class.

Properties

The IBMTS_NameSpace has the properties shown in Table 51.

Table 51. IBMTS_NameSpace properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description

Table 51. IBMTS_NameSpace properties (continued)

Property	Type	Qualifier	Description
ClassInfo	Uint16	Deprecated (CIM_Namespace), Required(TRUE), Write(TRUE), ModelCorrespondence (CIM_Namespace)	Identifies the organization of the namespace **Code** **Semantics** **0** Unknown **1** Other **2** CIM 1.0 **3** CIM 2.0 **4** CIM 2.1 **5** CIM 2.2 **6** CIM 2.3 **7** CIM 2.4 **8** CIM 2.5 **9** CIM 2.6 **10** CIM 2.7 **11** CIM 2.8 **200** DMI Recast **201** SNMP Recast **202** CMIP Recast
ClassType	Uint16	Write(TRUE), ModelCorrespondence (CIM_Namespace)	Indicates the schema of the namespace **Code** **Semantics** **0** Unknown **1** Other **2** CIM 1.0 **200** DMI Recast **201** SNMP Recast **202** CMIP Recast
ClassTypeVersion	String	Write(TRUE), ModelCorrespondence (CIM_Namespace)	Identifies the objects in the namespace
CreationClassName	String	MaxLen(256)	Indicates the name of the class or subclass used to create an instance
Description	String		Provides a textual description of the object
DescriptionOf ClassInfo	String	Deprecated (CIM_Namespace), Write(TRUE), ModelCorrespondence (CIM_Namespace)	Provides detailed information about the object
DescriptionOf ClassType	String	Write(TRUE), ModelCorrespondence (CIM_Namespace)	Provides detailed information about the object
ElementName	String		Specifies the name of the instance
Name	String	MaxLen(256)	Defines the unique label, in the context of the hosting system, by which the AccessControl Information is known

Table 51. IBMTS_NameSpace properties (continued)

Property	Type	Qualifier	Description
ObjectManager Creation	String	Propogated (CIM_ObjectManager), MaxLen(256)	Identifies the ObjectManager's CreationClassName
ObjectManagerName	String	Propogated (CIM_ObjectManager), MaxLen(256)	Identifies the ObjectManager's name
SystemCreationClass	String	Propogated (CIM_ObjectManager), MaxLen(256)	Indicates the system's creation class name
SystemName	String	Propogated (CIM_ObjectManager), MaxLen(256)	Identifies the system's name

IBMTS_ObjectManager

The IBMTS_ObjectManager class represents the Common Information Model Object Manager (CIMOM) itself.

Properties

The IBMTS_ObjectManager class extends the CIM_ObjectManager class and has the properties shown in Table 52.

Table 52. IBMTS_ObjectManager properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
Description	String		Provides a textual description of the object
ElementName	String	ReadRole(None)	Specifies the name of the instance
EnabledDefault	Uint16	Write(TRUE)	Indicates the administrator's default or startup configuration for an element's Enabled Status **Code** **Semantics** **2** Enabled **3** Disabled **5** Not Applicable **6** Enabled but Offline **7** No Default **8..32767** DMTF Reserved **32768...** Vendor Reserved

Table 52. IBMTS_ObjectManager properties (continued)

Property	Type	Qualifier	Description
EnabledState	Uint16	ModelCorrespondence (CIM_EnabledLog)	Indicates whether the element is currently shutting down or in an enabled or disabled state **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11..32767** DMTF Reserved **32768...** Vendor Reserved
GatherStatistical Data	Boolean	Write(TRUE)	Indicates whether the CIM_CIMOM StatisticalData object has gathered statistical data and whether the data is accessible
InstallDate	Date-time		Indicates the date on which the CIM client created the object in the CIMOM's repository
Name	String	MaxLen(256)	Identifies the Service and provides an indication of the functionality that is managed

Table 52. IBMTS_ObjectManager properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Indicates the operational status of the cluster **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Complete **18** Power Mode **..** DMTF Reserved **0x8000** Vendor Reserved
OtherEnabledStatus	String	ModelCorrespondence (CIM_EnabledLog)	Describes the element's enabled or disabled state when EnabledStatus is set to 1; must be null when EnabledStatus is set to a value other than 1
PrimaryOwner Contact	String	Write(TRUE), MaxLen(256)	Specifies how the primary owner of the account can be contacted, such as phone number or e-mail
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Specifies the name of the primary owner

Table 52. IBMTS_ObjectManager properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLog)	Sets the state (shut down, enabled, taken offline, or tested) of the element at the next operation **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10** Reboot **11** Reset **..** DMTF Reserved **32768...** Vendor Reserved
Started	Boolean		Indicates that the Service is started or stopped for the account
StartMode	String	MaxLen(10), Deprecated (CIM_Service.Enabled)	Indicates whether the Service is automatically started by a system, operating system, or specific user request
Status	String	MaxLen(10), Deprecated (CIM_ManagedSystem)	Indicates the current status of the Service instance
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Describes the status of the cluster when OperationalStatus is set to 1
SystemCreation ClassName	String	MaxLen(256)	Indicates the class creation name of the scoping system
SystemName	String	Propagated (CIM_System.Name), MaxLen(256)	Specifies the name of the scoping system
SystemCreation Class	String	Propagated (CIM_System.Class), MaxLen(256)	Indicates the class creation of the scoping system
TimeOfLast StateChange	Date-time		Indicated the date and time when the element's EnableState last changed
Version	String	Experimental(TRUE)	Indicates the VRMF level of the Common Information Model (CIM) Agent

IBMTS_RegisteredProfile

The IBMTS_RegisteredProfile extends the CIM_RegisteredProfile class.

Properties

The IBMTS_RegisteredProfile has the properties shown in Table 53.

Table 53. IBMTS_RegisteredProfile properties

Property	Type	Qualifier	Description
AdvertiseType Description	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_RegisteredProfile)	Provides information about the AdvertiseType
AdvertiseTypes	Uint16[]	Required(TRUE), ArrayType(Indexed), ModelCorrespondence (CIM_RegisteredProfile)	Signifies the advertisement for the profile information **Code** **Semantics** **1** Other **2** Not Advertise **3** SLP
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
Description	String		Provides a textual description of the object
ElementName	String		Specifies the name of the instance
InstanceID	String		Identifies an instance of this class
OtherRegistered Organization	String	ModelCorrespondence (CIM_ObjectManager), MaxLen(256)	Provides a descirption of the organization when 1 = Other is specified for the Registered Organization
RegisteredName	String	Required(TRUE), MaxLen(256)	Provides the name of this RegisteredProfile

Table 53. IBMTS_RegisteredProfile properties (continued)

Property	Type	Qualifier	Description
Registered Organization	Uint16	Required(TRUE), ModelCorrespondence (CIM_Namespace)	Identifies the organization of the namespace **Code** **Semantics** **1** Other **2** DMTF **3** CompTIA **4** Consortium for Service Innovation **5** FAST **6** GGF **7** INTAP **8** itSMF **9** NAC **10** Northwest Energy Efficiency Alliance **11** SNIA **12** TM Forum **13** The Open Group **14** ANSI **15** IEEE **16** IETF **17** INCITS **18** ISO **19** W3C
RegisteredVersion	String	Required(TRUE)	Indicates the version of this profile

IBMTS_System

The IBMTS_System extends the CIM_System class.

Properties

The IBMTS_System has the properties shown in Table 54.

Table 54. IBMTS_System properties

Property	Type	Qualifier	Description
Caption	String	MaxLen(64)	Identifies the object with a short (one-line string) textual description
CreationClassName	String	MaxLen(256)	Identifies the name of the class used in creation of an instance
Name	String	MaxLen(256)	Serves as key of a System instance
Description	String		Provides a textual description of the object
ElementName	String	ReadRole(None)	Specifies the name of the instance

Table 54. IBMTS_System properties (continued)

Property	Type	Qualifier	Description
EnabledDefault	Uint16	Write(TRUE)	Indicates the administrator's default or startup configuration for an element's Enabled Status. By default, the element is Enabled (value = 2) **Code** **Semantics** **0** Unknown **1** Other **2** Enabled **3** Disabled **4** Shutting Down **5** Not Applicable **6** Enabled but Offline **7** In Test **8** Deferred **9** Quiesce **10** Starting **11...32767** DMTF Reserved **32768...** Vendor Reserved
OtherEnabledState	Uint16	ModelCorrespondence (CIM_EnabledLog)	Indicates whether the element is currently shutting down or in an enabled or disabled state
InstallDate	Date-time		Indicates the date on which the CIM client created the object in the CIMOM's repository
NameFormat	String	MaxLen(64)	Provides the scope for numerous components

Table 54. IBMTS_System properties (continued)

Property	Type	Qualifier	Description
OperationalStatus	Uint16[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Indicates the operational status of the cluster **Code** **Semantics** **0** Unknown **1** Other **2** OK **3** Degraded **4** Stressed **5** Predictive Failure **6** Error **7** Non-Recoverable Error **8** Starting **9** Stopping **10** Stopped **11** In Service **12** No Contact **13** Lost Communication **14** Aborted **15** Dormant **16** Supporting Entity in Error **17** Complete **18** Power Mode **..** DMTF Reserved **0x8000** Vendor Reserved
PrimaryOwner Contact	String	MaxLen(256), Write(TRUE)	Specifies how the primary owner of the account can be contacted, such as phone number or e-mail
PrimaryOwner Name	String	MaxLen(64), Write(TRUE)	Specifies the name of the primary owner

Table 54. IBMTS_System properties (continued)

Property	Type	Qualifier	Description
RequestedState	Uint16	ModelCorrespondence (CIM_EnabledLog)	Sets the state (shut down, enabled, taken offline, or tested) of the element at the next operation **Code** **Semantics** **2** Enabled **3** Disabled **4** Shut Down **5** No Change **6** Offline **7** Test **8** Deferred **9** Quiesce **10** Reboot **11** Reset **..** DMTF Reserved **32768...** Vendor Reserved
Roles	String[]	Write(TRUE)	Specifies the administrator defined roles
Status	String	Deprecated (CIM_ManagedSystem), MaxLen(10)	Indicates the current status of the object
StatusDescriptions	String[]	ArrayType(Indexed), ModelCorrespondence (CIM_ManagedSystem)	Describes the various OperationalStatus array values
TimeOfLast StateChange	Date-time		Indicates the time the last change of state occurred

Association object classes

This section describes the association classes and their properties of the CIM Agent for the SAN Volume Controller.

IBMTSSVC_AllocatedFromStoragePool

The IBMTSSVC_AllocatedFromStoragePool class connects an IBMTSSVC_StorageVolume instance to an IBMTSSVC_StoragePool from which the volume is allocated.

References

The IBMTSSVC_AllocatedFromStoragePool class extends the CIM_AllocatedFromStoragePool class and has the references shown in Table 55 on page 229.

Table 55. IBMTSSVC_AllocatedFromStoragePool references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_StoragePool		The storage pool
Dependent	IBMTSSVC_Storage Volume		The storage volume

Properties

The IBMTSSVC_AllocatedFromStoragePool class has the properties shown in Table 56.

Table 56. IBMTSSVC_AllocatedFromStoragePool properties

Property	Type	Qualifier	Description
SpaceConsumed	Uint64	Units(Bytes), Required(TRUE), ModelCorrespondence (CIM_StoragePool.Total ManagedSpace CIM_StoragePool. RemainingManagedSpace)	Unsupported property

IBMTSSVC_AuthorizedCollection

The IBMTSSVC_AuthorizedCollection class associates a Host with a Privilege. AuthorizatedSubject is one of the elements in the authorization chain between a Host and a StorageVolume.

References

The IBMTSSVC_AuthorizedCollection class extends the CIM_AuthorizedSubject class and has the references shown in Table 57.

Table 57. IBMTSSVC_AuthorizedCollection references

Name	Reference	Qualifier	Description
Privilege	IBMTSSVC_Privilege		The Privilege
Privileged Element	IBMTSSVC_HardwareId Collection		The Host

IBMTSSVC_AuthorizedStorageHardwareID

The IBMTSSVC_AuthorizedStorageHardwareID class associates a Host with a Privilege. AuthorizatedSubject is one of the elements in the authorization chain between a Host and a StorageVolume.

References

The IBMTSSVC_AuthorizedStorageHardwareID class extends the CIM_AuthorizedSubject class and has the references shown in Table 58.

Table 58. IBMTSSVC_AuthorizedStorageHardwareID references

Name	Reference	Qualifier	Description
Privilege	IBMTSSVC_Privilege		The Privilege

Table 58. IBMTSSVC_AuthorizedStorageHardwareID references (continued)

Name	Reference	Qualifier	Description
Privileged Element	IBMTSSVC_Storage HardwareID		The StorageHardwareID

IBMTSSVC_AuthorizedSubject

The IBMTSSVC_AuthorizedSubject class associates a Host with a Privilege. AuthorizedSubject is one of the elements in the authorization chain between a Host and a Storage Volume.

References

The IBMTSSVC_AuthorizedSubject class extends the CIM_AuthorizedSubject class and has the references shown in Table 59.

Table 59. IBMTSSVC_AuthorizedSubject references

Name	Reference	Qualifier	Description
Privilege	IBMTSSVC_Privilege		The Privilege
Privilege Element	CIM_ManagedElement		The Host or StorageHardwareID

IBMTSSVC_AuthorizedTarget

The IBMTSSVC_AuthorizedTarget class associates an Privilege with a ProtocolController. AuthorizedTarget is one of the elements in the authorization chain between a StorageHardwareID and a StorageVolume.

References

The IBMTSSVC_AuthorizedTarget class extends the CIM_AuthorizedTarget class and has the references shown in Table 60.

Table 60. IBMTSSVC_AuthorizationTarget references

Name	Reference	Qualifier	Description
Privilege	IBMTSSVC_Privilege		The Privilege affecting the target resource
TargetElement	IBMTSSVC_Controller		The target set of resources to which the Privilege applies

IBMTSSVC_AvailableHardwareID

The IBMTSSVC_AvailableHardwareID class associates the AccountManagementService to CandidateStorageHardwareIDs.

References

The IBMTSSVC_AvailableHardwareID class extends the CIM_ConcreteDependency class and has the references shown in Table 61 on page 231.

Table 61. IBMTSSVC_AvailableHardwareID references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Storage HardwareIDManagement Service		The StorageHardware IDManagementService
Dependent	IBMTSSVC_Candidate StorageHardwareID		The CandidateStorage HardwareID

IBMTSSVC_BackendControllerForVolume

The IBMTSSVC_BackendControllerForVolume class associates BackendControllers with their Volumes.

References

The IBMTSSVC_BackendControllerForVolume class extends the CIM_Component class and has the references shown in Table 62.

Table 62. IBMTSSVC_BackendSCSILUN references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Backend Controller	Aggregate	The BackendController
Part Component	IBMTSSVC_Backend Volume		The BackendVolume controlled by the Antecedent

IBMTSSVC_BasedOn

The IBMTSSVC_BasedOn class associates StorageVolumes with the BackendVolumes where their data resides.

References

The IBMTSSVC_BasedOn class extends the CIM_BasedOn class and has the references shown in Table 63.

Table 63. IBMTSSVC_BasedOn references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Backend Volume		The BackendVolume containing data from the Dependent
Dependent	IBMTSSVC_Storage Volume		The StorageVolume

Properties

The IBMTSSVC_BasedOn class has the properties shown in Table 64.

Table 64. IBMTSSVC_BasedOn properties

Property	Type	Qualifier	Description
EndingAddress	Uint64		Unsupported property

Table 64. IBMTSSVC_BasedOn properties (continued)

Property	Type	Qualifier	Description
ExtentCount	Uint64		The number of extents that are allocated on the BackendVolume for the StorageVolume
OrderIndex	Uint64		Unsupported property
StartingAddress	Uint64		Unsupported property

IBMTSSVC_ClusterController

The IBMTSSVC_ClusterController class defines the Cluster scope of the Controller.

References

The IBMTSSVC_ClusterController class extends the CIM_SystemDevice class and has the references shown in Table 65.

Table 65. IBMTSSVC_ClusterController references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Cluster	Aggregate, Min, Max	The Cluster
Part Component	IBMTSSVC_Controller	Weak	The Controller

IBMTSSVC_ClusterDumps

The IBMTSSVC_ClusterDumps class extends the CIM_ElementSettingData class.

References

The IBMTSSVC_ClusterDumps class has the references shown in Table 66.

Table 66. IBMTSSVC_ClusterDumps references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Cluster		The Cluster
SettingData	IBMTSSVC_Dumps		The Dumps

Properties

The IBMTSSVC_ClusterDumps class has the properties shown in Table 67 on page 233.

Table 67. IBMTSSVC_ClusterDumps properties

Property	Type	Qualifier	Description
IsCurrent	Uint16		Indicates that the referenced setting is currently being used in the operation of the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Current **2** Is Not Current
IsDefault	Uint16		Indicates that the referenced setting is a default setting for the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Default **2** Is Not Default

IBMTSSVC_ClusteringCandidate

The IBMTSSVC_ClusteringCandidate class associates an IBMTSSVC_CandidateNode instance with an IBMTSSVC_Cluster instance.

References

The IBMTSSVC_ClusteringCandidate class extends the CIM_Dependency class and has the references shown in Table 68.

Table 68. IBMTSSVC_ClusteringCandidate references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Candidate Node		A node that is not a member of this or any other cluster
Dependent	IBMTSSVC_Cluster		The cluster

IBMTSSVC_ClusteringServiceForSystem

The IBMTSSVC_ClusteringServiceForSystem class associates an IBMTSSVC_Cluster instance with an IBMTSSVC_ClusteringService instance.

References

The IBMTSSVC_ClusteringServiceForSystem class extends the CIM_HostedClusterService class and has the references shown in Table 69.

Table 69. IBMTSSVC_ClusteringServiceForSystem references

Name	Target	Qualifier	Description
Antecedent	IBMTSSVC_Cluster	Min, Max	The Cluster

Table 69. IBMTSSVC_ClusteringServiceForSystem references (continued)

Name	Target	Qualifier	Description
Dependent	IBMTSSVC_Clustering Service	Weak	The ClusteringService that is hosted on the Cluster

IBMTSSVC_ClusterMaskingCapabilities

The IBMTSSVC_ClusterMaskingCapabilities class associates an IBMTSSVC_Cluster instance with its DeviceMaskingCapabilities.

References

The IBMTSSVC_ClusterMaskingCapabilities class extends the CIM_ElementCapabilities class and has the references shown in Table 70.

Table 70. IBMTSSVC_ClusterMaskingCapabilities references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Cluster	Min, Max	The IBMTSSVC_ ControllerMasking Capabilities for this cluster
Capabilities	IBMTSSVC_Controller MaskingCapabilities		The Capabilities object associated with the element

IBMTSSVC_ClusterPort

The IBMTSSVC_ClusterPort defines the cluster scope of the fibre-channel port.

References

The IBMTSSVC_ClusterPort class extends the CIM_SystemDevice class and has the references shown in Table 71.

Table 71. IBMTSSVC_ClusterPort references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Cluster	Aggregate, Min, Max	The Cluster
Part Component	IBMTSSVC_FCPort	Weak	The fibre-channel port

IBMTSSVC_ClusterScopeCandidateVolume

The IBMTSSVC_ClusterScopeCandidateVolume class associates IBMTSSVC_Cluster instance with an IBMTSSVC_CandidateVolume instance.

References

The IBMTSSVC_ClusterScopeCandidateVolume class extends the CIM_Dependency class and has the references shown in Table 72 on page 235.

Table 72. IBMTSSVC_ClusterScopeCandidateVolume references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_Candidate Volume		The CandidateVolume

IBMTSSVC_ClusterScopeChassis

The IBMTSSVC_ClusterScopeChassis class defines the cluster scope of the IBMTSSVC_Chassis instance.

References

The IBMTSSVC_ClusterScopeChassis class extends the CIM_Dependency class and has the references shown in Table 73.

Table 73. IBMTSSVC_ClusterScopeChassis references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_Chassis		The Chassis

IBMTSSVC_ClusterScopeFCSet

The IBMTSSVC_ClusterScopeFCSet class defines the cluster scope of an IBMTSSVC_FlashCopySynchronizedSet instance.

References

The IBMTSSVC_ClusterScopeFCSet class extends the CIM_Dependency class and has the references shown in Table 74.

Table 74. IBMTSSVC_ClusterScopeFCSet references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_FlashCopy SynchronizedSet		The FlashCopy SynchronizedSet instance

IBMTSSVC_ClusterScopeIOGroup

The IBMTSSVC_ClusterScopeIOGroup defines the cluster scope of the I/O group.

References

The IBMTSSVC_ClusterScopeIOGroup class extends the CIM_Dependency class and has the references shown in Table 75.

Table 75. IBMTSSVC_ClusterScopeIOGroup references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_IOGroup		The I/O group

IBMTSSVC_ClusterScopeNodeVPD

The IBMTSSVC_ClusterScopeNodeVPD class defines the cluster scope of an IBMTSSVC_NodeVPD instance.

References

The IBMTSSVC_ClusterScopeNodeVPD class extends the CIM_Dependency class and has the references shown in Table 76.

Table 76. IBMTSSVC_ClusterScopeNodeVPD references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_NodeVPD		The NodeVPD instance

IBMTSSVC_ClusterScopePrivilege

The IBMTSSVC_ClusterScopePrivilege class defines the cluster scope of an IBMTSSVC_Privilege instance.

References

The IBMTSSVC_ClusterScopePrivilege class extends the CIM_Dependency class and has the references shown in Table 77.

Table 77. IBMTSSVC_ClusterScopePrivilege references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_Privilege		The Privilege

IBMTSSVC_ClusterScopeProduct

The IBMTSSVC_ClusterScopeProduct class defines the cluster scope of an IBMTSSVC_Product instance.

References

The IBMTSSVC_ClusterScopeProduct class extends the CIM_Dependency class and has the references shown in Table 78.

Table 78. IBMTSSVC_ClusterScopeProduct references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_Product		The Product instance

IBMTSSVC_ClusterScopeSCSet

The IBMTSSVC_ClusterScopeSCSet class defines the Cluster scope of an IBMTSSVC_SyncCopySynchronizedSet instance.

References

The IBMTSSVC_ClusterScopeSCSet class extends the CIM_Dependency class and has the references shown in Table 79 on page 237.

Table 79. IBMTSSVC_ClusterScopeSCSet references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_SyncCopy SynchronizedSet		The SyncCopySynchronized Set instance

IBMTSSVC_ClusterVolume

The IBMTSSVC_ClusterVolume class defines the cluster scope of an IBMTSSVC_StorageVolume instance.

References

The IBMTSSVC_ClusterScopeVolume class extends the CIM_Dependency class and has the references shown in Table 80.

Table 80. IBMTSSVC_ClusterVolume references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_Storage Volume		The StorageVolume

IBMTSSVC_ComponentCS

The IBMTSSVC_ComponentCS class associates the cluster and its nodes.

References

The IBMTSSVC_ComponentCS class extends the CIM_ComponentCS class and has the references shown in Table 81.

Table 81. IBMTSSVC_ComponentCS references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Cluster	Aggregate	The Cluster
Part Component	IBMTSSVC_Node		The Node

IBMTSSVC_ComputerSystemPackage

The IBMTSSVC_ComputerSystemPackage class connects an IBMTSSVC_Node instance with the corresponding IBMTSSVC_Chassis instance.

References

The IBMTSSVC_ComputerSystemPackage class extends the CIM_ComputerSystemPackage class and has the references shown in Table 82 on page 238.

Table 82. IBMTSSVC_ComputerSystemPackage references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Chasis		The Chassis
Dependent	IBMTSSVC_Node		The Node

Properties

The IBMTSSVC_ComputerSystemPackage class has the properties shown in Table 83.

Table 83. IBMTSSVC_ComputerSystemPackage properties

Property	Type	Qualifier	Description
PlatformGUID	String		Unsupported property.

IBMTSSVC_ConnectedBackendController

The IBMTSSVC_ConnectedBackendController class connects an IBMTSSVC_Cluster instance to an IBMTSSVC_BackendController instance that is visible in the fibre-channel SAN.

References

The IBMTSSVC_ConnectedBackendController class extends the CIM_Dependency class and has the references shown in Table 84.

Table 84. IBMTSSVC_ConnectedBackendController references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_Backend Controller		The Backend Controller that is connected to the Cluster

IBMTSSVC_ControllerConfigurationServiceForSystem

The IBMTSSVC_ControllerConfigurationServiceForSystem class connects an IBMTSSVC_Cluster instance to the corresponding IBMTSSVC_ControllerConfigurationService instance.

References

The IBMTSSVC_ControllerConfigurationServiceForSystem class extends the CIM_HostedService class and has the references shown in Table 85.

Table 85. IBMTSSVC_ControllerConfigurationServiceForSystem references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster	Min, Max	The system that the service should be used for
Dependent	IBMTSSVC_Controller ConfigurationService	Weak	The Controller Configuration Service that provides services for the system

IBMTSSVC_ControllerConfServiceMaskingCapabilities

The IBMTSSVC_ControllerConfServiceMaskingCapabilities class associates an IBMTSSVC_ControllerConfService instance with its DeviceMaskingCapabilities.

References

The IBMTSSVC_ControllerConfServiceMaskingCapabilities class extends the CIM_ElementCapabilities class and has the references shown in Table 86.

Table 86. IBMTSSVC_ControllerConfService MaskingCapabilities references

Name	Reference	Qualifier	Description
Managed Element	IBMTSSVC_Controller ConfigurationService	Min, Max	The IBMTSSVC_ ControllerMasking Capabilities for this service
Capabilities	IBMTSSVC_Controller MaskingCapabilities		The Capabilities object associated with the element

IBMTSSVC_CopyCandidate

The IBMTSSVC_CopyCandidate class associates an IBMTSSVC_CandidateVolume instance with an IBMTSSVC_StorageVolume instance, both of which must have the same characteristics.

References

The IBMTSSVC_CopyCandidate class extends the CIM_Dependency class and has the references shown in Table 87.

Table 87. IBMTSSVC_CopyCandidate references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Candidate Volume	Key	The Candidate Volume that is a potential auxiliary for a sync copy relationship with the Storage Volume
Dependent	IBMTSSVC_Storage Volume	Key	The Storage Volume

IBMTSSVC_ElementConformsToProfile

The IBMTSSVC_ElementConformsToProfile extends the CIM_ElementConformsToProfile class.

References

The IBMTSSVC_ElementConformsToProfile class has the references shown in Table 88 on page 240.

Table 88. IBMTSSVC_ComponentCS references

Name	Reference	Qualifier	Description
Conformant Standard	IBMTSSVC_Registered Profile		The Registered Profile to which the Managed Element conforms
Managed Element	IBMTSSVC_Cluster		The Managed Element that conforms to the Registered Profile

IBMTSSVC_FlashCopyStorageSynchronized

The IBMTSSVC_FlashCopyStorageSynchronized class aggregates a source IBMTSSVC_StorageVolume instance and a target IBMTSSVC_StorageVolume for a FlashCopy relationship.

References

The source and target volumes can reside on different IBMTSSVC_RedundancyGroup instances, but must be managed by the same IBMTSSVC_Cluster instance. The IBMTSSVC_FlashCopyStorageSynchronized class extends the CIM_StorageSynchronized class and has the references shown in Table 89.

Table 89. IBMTSSVC_FlashCopyStorageSynchronized references

Name	Reference	Qualifier	Description
SystemElement	IBMTSSVC_Storage Volume	MappingStrings	The Storage Volume that is the source for the replication
SyncedElement	IBMTSSVC_Storage Volume	MappingStrings	The Storage Volume that is the target for the replication

Properties

The IBMTSSVC_FlashCopyStorageSynchronized class has the properties shown in Table 90.

Table 90. IBMTSSVC_FlashCopyStorageSynchronized properties

Property	Type	Qualifier	Description
CopyRate	Uint16	Write(TRUE), WriteRole (Administrator)	Specifies the copy rate (%) for the SAN Volume Controller

Table 90. IBMTSSVC_FlashCopyStorageSynchronized properties (continued)

Property	Type	Qualifier	Description
CopyType	Uint16		The Replication Policy. **Code** **Semantics** **2** Async. Create and maintain an asynchronous copy of the source. **3** Sync. Create and maintain a synchronized copy of the source. **4** UnSyncAssoc. Create an un-synchronized copy and maintain an association to the source. **..** DMTF reserved **0x8000..** Vendor specific
ElementName	String	Write(TRUE), WriteRole (Administrator)	The user-friendly name of the association
Name	String		The name of the association
Progress	Uint32	Units(percent)	The status of the ongoing copy process
ReplicaType	Uint16		The type of replication relationship. The SAN Volume Controller replicas are FullCopy (0). **Code** **Semantics** **0** FullCopy **1** BeforeDelta **2** AfterDelta **3** Log **4** NotSpecified **..** DMTF reserved **0x8000..** Vendor specific
SyncedElementName	String		The name of the Synced Element
SynchronizedSet	String		The name of the SynchronizedSet with which the StorageSynchronized is associated

Table 90. IBMTSSVC_FlashCopyStorageSynchronized properties (continued)

Property	Type	Qualifier	Description
SynchronizedSetID	String		The identifier of the SynchronizedSet with which the StorageSynchronized is associated
SyncMaintained	Boolean		Indicates whether synchronization is maintained
SyncState	Uint16		The state of the synchronization **Code** **Semantics** **2** Initialized **3** PrepareIn Progress **4** Prepared **5** ResyncIn Progress **11** Idle **12** Broken **0x8000** Stopped
SystemElementName	String		The name of the SystemElement
WhenSynced	Date-time		Unsupported property

IBMTSSVC_FlashCopySynchronizedMember

The IBMTSSVC_FlashCopySynchronizedMember class associates an IBMTSSVC_FlashCopySynchronizedSet with its members, which are multiple IBMTSSVC_FlashCopySynchronization instances.

References

The IBMTSSVC_FlashCopySynchronizedMember class extends the CIM_SynchronizedMember class and has the references shown in Table 91.

Table 91. IBMTSSVC_FlashCopySynchronizedMember references

Name	Reference	Qualifier	Description
Collection	IBMTSSVC_FlashCopy SynchronizedSet	Aggregate	The FlashCopySynchronized Set
Member	IBMTSSVC_FlashCopy StorageSynchronized		The member of the set

IBMTSSVC_HardwareIDOnSystem

The IBMTSSVC_HardwareIDOnSystem associates a cluster with its storage hardware IDs.

References

The IBMTSSVC_HardwareIDOnSystem class extends the CIM_Dependency class and has the references shown in Table 92.

Table 92. IBMTSSVC_HardwareIDOnSystem references

Name	Reference	Qualifier	Description
Dependent	IBMTSSVC_Cluster		The Cluster
Antecedent	IBMTSSVC_Storage HardwareID		The Storage Hardware ID

IBMTSSVC_HostedAccessPoint

The IBMTSSVC_HostedAccessPoint extends the CIM_HostedAccessPoint class.

References

The IBMTSSVC_HostedAccessPoint class has the references shown in Table 93.

Table 93. IBMTSSVC_HostedAccessPoint references

Name	Reference	Qualifier	Description
Dependent	IBMTSSVC_System	Min, Max	The hosting system
Antecedent	IBMTSSVC_Remote ServiceAccessPoint	Weak	The SAP(s) that are hosted on this system

IBMTSSVC_HostedJob

The IBMTSSVC_HostedJob class associates an IBMTSSVC_Job instance with the IBMTSSVC_Cluster instance on which the job is running.

References

The IBMTSSVC_HostedJob class extends the CIM_Dependency class and has the references shown in Table 94.

Table 94. IBMTSSVC_HostedJob references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster	Key	The Cluster
Dependent	IBMTSSVC_Job	Key	The Job

IBMTSSVC_HostedPrimordialPool

The IBMTSSVC_HostedPrimordialPool associates a cluster with its primordial storage pools.

References

The IBMTSSVC_HostedPrimordialPool class extends the CIM_HostedStoragePool class and has the references shown in Table 95 on page 244.

Table 95. IBMTSSVC_HostedPrimordialPool references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Cluster	Aggregate, Max, Min	The Cluster
Part Component	IBMTSSVC_Primordial StoragePool		The Primordial Storage Pool

IBMTSSVC_HostedStoragePool

The IBMTSSVC_HostedStoragePool class connects an IBMTSSVC_Cluster instance with the corresponding IBMTSSVC_StoragePool instance.

References

The IBMTSSVC_HostedStoragePool class extends the CIM_HostedStoragePool class and has the references shown in Table 96.

Table 96. IBMTSSVC_HostedStoragePool references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Cluster	Min, Max, Aggregate	The Cluster
Part Component	IBMTSSVC_StoragePool		The StoragePool

IBMTSSVC_HwIDCollectionOnSystem

The IBMTSSVC_HwIDCollectionOnSystem associates a cluster with its hardware ID.

References

The IBMTSSVC_HwIDCollectionOnSystem class extends the CIM_Dependency class and has the references shown in Table 97.

Table 97. IBMTSSVC_HwIDCollectionOnSystem references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The Cluster
Dependent	IBMTSSVC_HardwareId Collection		The hardware ID

IBMTSSVC_IndicationFiltersConformsToProfile

The IBMTSSVC_IndicationFiltersConformsToProfile class associates IndicationFilters and the InBand profile.

References

The IBMTSSVC_IndicationFiltersConformsToProfile class extends the CIM_ElementConformsToProfile class and has the references shown in Table 98 on page 245.

Table 98. IBMTSSVC_IndicationFiltersConformsToProfile references

Name	Reference	Qualifier	Description
Conformant Standard	IBMTSSVC_ RegisteredProfile		The RegisteredProfile to which the ManagedElement conforms
ManagedElement	IBMTS_IndicationFilter		The IndicationFilter that conforms to the RegisteredProfile

IBMTSSVC_IndicationFiltersConformsToSubProfile

The IBMTSSVC_IndicationFiltersConformsToSubProfile class associates IndicationFilters and the InBand subprofile.

References

The IBMTSSVC_IndicationFiltersConformsToSubProfile class extends the CIM_ElementConformsToProfile class and has the references shown in Table 99.

Table 99. IBMTSSVC_IndicationFiltersConformsToSubProfile references

Name	Reference	Qualifier	Description
Conformant Standard	IBMTSSVC_ RegisteredSubProfile		The RegisteredSub Profile to which the ManagedElement conforms
Managed Element	IBMTS_IndicationFilter		The IndicationFilter that conforms to the RegisteredProfile

IBMTSSVC_IOGroupIdentity

The IBMTSSVC_IOGroupIdentity class extends the CIM_ConcreteIdentity class.

References

The IBMTSSVC_IOGroupIdentity class has the references shown in Table 100.

Table 100. IBMTSSVC_IOGroupIdentity references

Name	Reference	Qualifier	Description
SystemElement	IBMTSSVC_IOGroup		An aspect of the ManagedElement.
SameElement	IBMTSSVC_IOGroupSet		An aspect of the ManagedElement

IBMTSSVC_IOGroupPort

The IBMTSSVC_IOGroupPort associates an I/O group with its Fibre Channel ports.

References

The IBMTSSVC_IOGroupPort class extends the CIM_SystemDevice class and has the references shown in Table 101 on page 246.

Table 101. IBMTSSVC_IOGroupPort references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_IOGroup	Aggregate, Max, Min	The parent system in the Association
Part Component	IBMTSSVC_FCPort	Weak	The LogicalDevice that is a component of a System

IBMTSSVC_ManagesCollection

The IBMTSSVC_ManagesCollection associates a HardwareIdCollection with the StorageHardwareIDManagementService to indirectly manage it.

References

The IBMTSSVC_ManagesCollection class extends the CIM_ConcreteDependency class and has the references shown in Table 102.

Table 102. IBMTSSVC_ManagesCollection references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Storage HardwareIDManagement Service		The StorageHardwareID ManagementService
Dependent	IBMTSSVC_Hardware IdCollection		The hardware ID collection

IBMTSSVC_ManagesController

The IBMTSSVC_ManagesController class associates a Controller with the ControllerConfigurationService to manage it.

References

The IBMTSSVC_ManagesController class extends the CIM_ConcreteDependency class and has the references shown in Table 103.

Table 103. IBMTSSVC_ManagesController references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Controller ConfigurationService		The ControllerConfiguration Service
Dependent	IBMTSSVC_Controller		The Controller

IBMTSSVC_ManagesHardwareID

The IBMTSSVC_ManagesHardwareID class associates a HardwareID with the StorageHardwareIDManagementService to manage it.

References

The IBMTSSVC_ManagesHardwareID class extends the CIM_ConcreteDependency class and has the references shown in Table 104 on page 247.

Table 104. IBMTSSVC_ManagesHardwareID references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Storage HardwareID ManagementService		The StorageHardware IDManagementService
Dependent	IBMTSSVC_Controller StorageHardwareID		The StorageHardwareID

IBMTSSVC_ManagesPrivilege

The IBMTSSVC_ManagesPrivilege class associates a Privilege with the LunMaskPrivilegeServiceto manage it.

References

The IBMTSSVC_ManagesPrivilege class extends the CIM_ConcreteDependency class and has the references shown in Table 105.

Table 105. IBMTSSVC_ManagesPrivilege references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Privilege ManagementService		The Privilege Managment Service
Dependent	IBMTSSVC_Privilege		The Privilege

IBMTSSVC_MemberOfCollection

The IBMTSSVC_MemberOfCollection class associates a Host with its StorageHardwareID (fibre-channel ports).

References

The IBMTSSVC_MemberOfCollection class extends the CIM_MemberOfCollection class and has the references shown in Table 106.

Table 106. IBMTSSVC_MemberOfCollection references

Name	Reference	Qualifier	Description
Collection	IBMTSSVC_Hardware IdCollection	Aggregate	The Host
Member	IBMTSSVC_Storage HardwareID		The Storage Hardware ID (fibre-channel port)

IBMTSSVC_MemberOfIOGroup

The IBMTSSVC_MemberOfIOGroup class associates a node with the IOGroupSet to which it belongs.

References

The IBMTSSVC_MemberOfIOGroup class extends the CIM_MemberOfCollection class and has the references shown in Table 107 on page 248.

Table 107. IBMTSSVC_MemberOfIOGroup references

Name	Reference	Qualifier	Description
Collection	IBMTSSVC_IOGroupSet	Aggregate	The RedundancyGroup
Member	IBMTSSVC_Node		The Node

IBMTSSVC_NodeDumps

The IBMTSSVC_NodeDumps class shows the dumps found on a specific node.

References

The IBMTSSVC_NodeDumps class extends the CIM_ElementSettingData class and has the references shown in Table 108.

Table 108. IBMTSSVC_NodeDumps references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Node		The Node
SettingData	IBMTSSVC_Dumps		The Dumps

Properties

The IBMTSSVC_NodeDumps class has the properties shown in Table 109.

Table 109. IBMTSSVC_NodeDumps

Name	Type	Qualifier	Description
IsCurrent	Uint16		An enumerated integer that indicates that the referenced setting is currently being used in the operation of the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Current **2** Is Not Current
IsDefault	Uint16		An enumerated integer that indicates that the referenced setting is a default setting for the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Default **2** Is Not Default

IBMTSSVC_PartnershipCandidate

The IBMTSSVC_PartnershipCandidate class associates the Cluster with the available remote clusters.

References

The IBMTSSVC_PartnershipCandidate class extends the CIM_Dependency class and has the references shown in Table 110.

Table 110. IBMTSSVC_PartnershipCandidate references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The local Cluster.
Dependent	IBMTSSVC_Candidate Cluster		The remote Cluster.

IBMTSSVC_PoolCapabilities

The IBMTSSVC_PoolCapabilities class associates a storage pool with a storage capabilities instance.

References

The IBMTSSVC_PoolCapabilities class extends the CIM_ElementCapabilities class and has the references shown in Table 111.

Table 111. IBMTSSVC_PoolCapabilities references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_StoragePool	Min, Max	The managed element
Capabilities	IBMTSSVC_Storage Capabilities		The Capabilities object associated with the element

IBMTSSVC_PrimordialPoolCapabilities

The IBMTSSVC_PrimordialPoolCapabilities class extends the CIM_ElementCapabilities class.

References

The IBMTSSVC_PrimordialPoolCapabilities class has the references shown in Table 112.

Table 112. IBMTSSVC_PrimordialPoolCapabilities references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Primordial StoragePool	Min, Max	The managed element
Capabilities	IBMTSSVC_Storage Capabilities		The Capabilities object associated with the element

IBMTSSVC_PrimordialPoolComponent

The IBMTSSVC_PrimordialPoolComponent class associates a PrimordialPool with the BackendVolumes from which it is assembled.

References

The IBMTSSVC_PrimordialPoolComponent class extends the CIM_ConcreteComponent class and has the references shown in Table 113.

Table 113. IBMTSSVC_PrimordialPoolComponent references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Primordial StoragePool	Aggregate	The Primordial StoragePool
Part Component	IBMTSSVC_Backend Volume		The BackendVolume

IBMTSSVC_PrimordialPoolForController

The IBMTSSVC_PrimordialPoolForController class associates a BackendController with the corresponding PrimordialPool.

References

The IBMTSSVC_PrimordialPoolForController class extends the CIM_Dependency class and has the references shown in Table 114.

Table 114. IBMTSSVC_PrimordialPoolForController references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Backend Controller		The BackendController
Dependent	IBMTSSVC_Primordial StoragePool		The primordial storage pool

IBMTSSVC_PrivilegeServiceForSystem

The IBMTSSVC_PrivilegeServiceForSystem class associates a Cluster with its PrivilegeManagementService.

References

The IBMTSSVC_PrivilegeServiceForSystem class extends the CIM_HostedService class and has the references shown in Table 115.

Table 115. IBMTSSVC_PrivilegeServiceForSystem references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster	Min, Max	The system for which the service should be used
Dependent	IBMTSSVC_Privilege ManagementService	Weak	The Privilege Management Service that provides services for the system

IBMTSSVC_ProductPhysicalComponent

The IBMTSSVC_ProductPhysicalComponent class associates an IBMTSSVC_Product instance with the corresponding IBMTSSVC_Chassis instance.

References

The IBMTSSVC_ProductPhysicalComponent class extends the CIM_ProductPhysicalComponent class and has the references shown in Table 116.

Table 116. IBMTSSVC_ProductPhysicalComponent references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Product	Max, Aggregate	The Product
Part Component	IBMTSSVC_Chassis		The Chassis

IBMTSSVC_ProtocolControllerForPort

The IBMTSSVC_ProtocolControllerForPort class associates a Controller with the fibre-channel ports through which it can be accessed.

References

A Controller instance has a corresponding RedundancyGroup. The RedundancyGroup associates one or two nodes, and a node associates fibre-channel ports. ControllerFCPort provides a shortcut for traversing from a Controller to the fibre-channel ports.

The IBMTSSVC_ProtocolControllerForPort class extends the CIM_ProtocolControllerForPort class and has the references shown in Table 117.

Table 117. IBMTSSVC_ProtocolControllerForPort references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Controller	Experimental	The protocol controller
Dependent	IBMTSSVC_FCPort	Experimental	The port

Properties

The IBMTSSVC_ProtocolControllerForPort class has the properties shown in Table 118.

Table 118. IBMTSSVC_ProtocolControllerForPort

Name	Type	Qualifier	Description
AccessPriority	Uint16	Experimental(TRUE)	The priority given to accesses of the device through this controller. The highest priority path will have the lowest value for this parameter. No priorities, constant value of 0.

Table 118. IBMTSSVC_ProtocolControllerForPort (continued)

Name	Type	Qualifier	Description
AccessState	Uint16	Experimental(TRUE)	Indicates whether the Controller is actively commanding or accessing the Device. This information is necessary when a LogicalDevice can be commanded by or accessed through multiple Controllers. **Code** **Semantics** **0** Unknown **1** Active **2** Inactive
DeviceNumber	String	Experimental(TRUE)	Address of the associated Device in the context of the antecedent Controller. Because the port has no specialid in the controller's context, this is a constant value of 0.

IBMTSSVC_ProtocolControllerForUnit

The IBMTSSVC_ProtocolControllerForUnit class associates a StorageVolume with Controller instances through which it is exposed to clients.

References

The IBMTSSVC_ProtocolControllerForUnit class extends the CIM_ProtocolControllerForUnit class and has the references shown in Table 119.

Table 119. IBMTSSVC_ProtocolControllerForUnit references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Controller	Experimental	The protocol controller
Dependent	IBMTSSVC_Storage Volume	Experimental	The volume

Properties

The IBMTSSVC_ProtocolControllerForUnit class has the properties shown in Table 120 on page 253.

Table 120. IBMTSSVC_ProtocolControllerForUnit

Property	Type	Qualifier	Description
AccessPriority	Uint16	Experimental(TRUE)	The priority given to accesses of the device through this controller. The highest priority path will have the lowest value for this parameter. No priorities, constant value of 0.
AccessState	Uint16	Experimental(TRUE)	Indicates whether the Controller is actively commanding or accessing the Device. This information is necessary when a Logical Device can be commanded by or accessed through multiple Controllers. **Code** **Semantics** **0** Unknown **1** Active **2** Inactive
DeviceNumber	String	Experimental(TRUE)	Address of the associated Device in the context of the antecedent Controller. This is the LUN number.
UniqueID	String		The unique ID of the volume shown on SCSI inquiry.

IBMTSSVC_ProviderInObjectManager

The IBMTSSVC_ProviderInObjectManager class associates the CIM Object Manager with its providers.

References

The IBMTSSVC_ProviderInObjectManager class extends the CIM_Component class and has the references shown in Table 121.

Table 121. IBMTSSVC_ProviderInObjectManager references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Object Manager	Aggregate	
Part Component	IBMTSSVC_Provider		

IBMTSSVC_RemotePartnership

The IBMTSSVC_RemotePartnership class associates the Cluster with the selected remote clusters.

References

The IBMTSSVC_RemotePartnership class extends the CIM_Dependency class and has the references shown in Table 122.

Table 122. IBMTSSVC_RemotePartnership references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster		The local Cluster
Dependent	IBMTSSVC_Remote Cluster		The remote Cluster

IBMTSSVC_RemoteSystemVolume

The IBMTSSVC_RemoteSystemVolume class associates the IBMTS_RemoteCluster instance with potential IBMTSSVC_CandidateVolumes.

References

The IBMTSSVC_RemoteSystemVolume class extends the CIM_Component class and has the references shown in Table 123.

Table 123. IBMTSSVC_RemoteSystemVolume references

Name	Target	Qualifier	Description
Group Component	IBMTSSVC_Remote Cluster	Aggregate	The Remote Cluster
PartComponent	IBMTSSVC_Remote Volume		The potential sync copy auxiliary volume.

IBMTSSVC_RequiresProfile

The IBMTSSVC_RequiresProfile class extends the CIM_SubProfileRequiresProfile class.

References

The IBMTSSVC_RequiresProfile class has the references shown in Table 124.

Table 124. IBMTSSVC_RequiresProfile references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Registered Profile	Min	The Registered Profile that is referenced or required by the subprofile.
Dependent	IBMTSSVC_Registered SubProfile		A Registered SubProfile that requires a scoping profile for context.

IBMTSSVC_SAPAvailableForElement

The IBMTSSVC_SAPAvailableForElement class associates a service access point with the device for which it offers a management interface.

References

The IBMTSSVC_SAPAvailableForElement class extends the CIM_SAPAvailableForElement class and has the references shown in Table 125.

Table 125. IBMTSSVC_SAPAvailableForElement references

Name	Reference	Qualifier	Description
AvailableSAP	IBMTSSVC_Remote ServiceAccessPoint	Min	The Service Access Point that is available.
ManagedElement	IBMTSSVC_Cluster		The ManagedElement for which the Service Access Point is available.

IBMTSSVC_StorageConfigurationServiceCapabilities

The IBMTSSVC_StorageConfigurationServiceCapabilities class associates an instance of IBMTSSVC_StorageConfigurationService with its DeviceMaskingCapabilities.

References

The IBMTSSVC_StorageConfigurationServiceCapabilities class extends the CIM_ElementCapabilities class and has the references shown in Table 126.

Table 126. IBMTSSVC_StorageConfigurationServiceCapabilities references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Storage ConfigurationService	Min, Max	The IBMTSSVC_Storage ConfigurationService for this cluster
SettingData	IBMTSSVC_Storage ConfigurationCapabilities		The Capabilities object associated with the element

IBMTSSVC_StorageConfigurationServiceForSystem

The IBMTSSVC_StorageConfigurationServiceForSystem class associates an IBMTSSVC_Cluster instance with its corresponding IBMTSSVC_StorageConfigurationService instance.

References

The IBMTSSVC_StorageConfigurationServiceForSystem class extends the CIM_HostedService class and has the references shown in Table 127.

Table 127. IBMTSSVC_StorageConfigurationServiceForSystem references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster	Min, Max	The Cluster
Dependent	IBMTSSVC_Storage ConfigurationService	Weak	The Storage ConfigurationService

IBMTSSVC_StorageHardwareIDManagementServiceForSystem

The IBMTSSVC_StorageHardwareIDManagementServiceForSystem class associates a Cluster with its StorageHardwareIDManagementService.

References

The IBMTSSVC_StorageHardwareIDManagementServiceForSystem class extends the CIM_HostedService class and has the references shown in Table 128.

Table 128. IBMTSSVC_StorageHardwareIDManagementServiceForSystem references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_Cluster	Min, Max	The system for which the service should be used
Dependent	IBMTSSVC_Storage HardwareIDManagement Service	Weak	The StorageHardware IDManagementService that provides services for the system

IBMTSSVC_StoragePoolComponent

The IBMTSSVC_StoragePoolComponent class associates the IBMTSSVC_StoragePool instances to the IBMTSSVC_BackendVolume instances from which the StoragePool is assembled.

References

The IBMTSSVC_StoragePoolComponent class extends the CIM_ConcreteComponent class and has the references shown in Table 129.

Table 129. IBMTSSVC_StoragePoolComponent references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_StoragePool	Aggregate	The storage pool
Part Component	IBMTSSVC_Backend Volume		The back-end volume

IBMTSSVC_SyncCopyStorageSynchronized

The IBMTSSVC_SyncCopyStorageSynchronized class associates an IBMTSSVC_StorageVolume instance with an IBMTSSVC_CandidateVolume or another IBMTSSVC_StorageVolume instance for a synchronous copy relationship.

References

The IBMTSSVC_SyncCopyStorageSynchronized class extends the CIM_StorageSynchronized class and has the references shown in Table 130.

Table 130. IBMTSSVC_SyncCopyStorageSynchronized references

Name	Reference	Qualifier	Description
SystemElement	CIM_LogicalElement	MappingStrings	The StorageVolume that is the master in the relationship

Table 130. IBMTSSVC_SyncCopyStorageSynchronized references (continued)

Name	Reference	Qualifier	Description
SyncedElement	CIM_LogicalElement	MappingStrings	The StorageVolume that is the auxiliary in the relationship

Properties

The IBMTSSVC_SyncCopyStorageSynchronized class has the properties shown in Table 131.

Table 131. IBMTSSVC_SyncCopyStorageSynchronized properties

Property	Type	Qualifier	Description
BackgroundCopy Priority	Uint16	Write(TRUE), WriteRole(Administrator)	The background copy priority, in the range 1 to 100. The default is 50.
Connected	Boolean		The status of the connection between the StorageVolumes
CopyType	Uint16		The Replication Policy. **Code** **Semantics** **2** Async. Create and maintain an asynchronous copy of the source. **3** Sync. Create and maintain a synchronized copy of the source. **4** UnSyncAssoc. Create an un-synchronized copy and maintain an association to the source. **..** DMTF Reserved **0x8000..** Vendor Specific
ElementName	String	Write(TRUE), WriteRole(Administrator)	The user-friendly name of this association
FreezeTime	String		The time when the copy relationship was removed
Name	String		The name of the association

Table 131. IBMTSSVC_SyncCopyStorageSynchronized properties (continued)

Property	Type	Qualifier	Description
NativeState	Uint16		The native state of the copy relationship **Code** **Semantics** **0** Idling **1** Idling disconnected **2** Consistent synchronized **3** Consistent disconnected **4** Consistent stopped **5** Inconsistent copying **6** Inconsistent disconnected **7** Inconsistent stopped
Primary	Uint32		Indicates which of the StorageVolumes is currently the primary in the copy relationship. The primary volume is the one accessible for I/O by the clients. **Code** **Semantics** **0** Master **1** Auxiliary
Progress	Uint32	Units (Percent)	The progress of the copy process, if one is ongoing
ReplicaType	Uint16		The type of the replica. SAN Volume Controller replicas are always FullCopy (0). **Code** **Semantics** **0** FullCopy **1** BeforeDelta **2** AfterDelta **3** Log **4** NotSpecified **..** DMTF Reserved **0x8000..** Vendor Specific
Status			The status of the relationship. **Code** **Semantics** **0** Online **1** Primary Offline **2** Secondary Offline

Table 131. IBMTSSVC_SyncCopyStorageSynchronized properties (continued)

Property	Type	Qualifier	Description
SyncedElement ClusterID	String		The ID of the SyncedElement's cluster
SyncedElement ClusterName	String		The name of the SyncedElement's cluster
SyncedElementID	String		The ID of the SyncedElement
SyncedElementName	String		The name of the SyncedElement
SynchronizedSet	String		The name of the SynchronizedSet with which the StorageSynchronized is associated
SynchronizedSetID	String		The ID of the SynchronizedSet with which the StorageSynchronized is associated
SyncMaintained	Boolean		Indicates whether synchronization is maintained
SyncState	Uint16		The state of the synchronization **Code** **Semantics** **4** Prepared **5** ReSync InProgress **6** Synchronized **12** Broken **13** Fractured **0x8101** Fractured Idle
SystemElement ClusterID	String		The ID of the SystemElement's cluster
SystemElement ClusterName	String		The name of the SystemElement's cluster
SystemElementID	String		The ID of the SystemElement
SystemElementName	String		The name of the SystemElement
WhenSynced	Date-time		Unsupported property

IBMTSSVC_SyncCopySynchronizedMember

The IBMTSSVC_SyncCopySynchronizedMember class associates an IBMTSSVC_FlashCopySynchronizedSet instance with its member IBMTSSVC_SyncCopySynchronized instances.

References

The IBMTSSVC_SyncCopySynchronizedMember class extends the CIM_SynchronizedMember class and has the references shown in Table 132.

Table 132. IBMTSSVC_SyncCopySynchronizedMember references

Name	Reference	Qualifier	Description
Collection	IBMTSSVC_SyncCopy SynchronizedSet	Aggregate	The SyncCopy SynchronizedSet
Member	IBMTSSVC_SyncCopy StorageSynchronized		The aggregated member of the set

IBMTSSVC_SystemBackendVolume

The IBMTSSVC_SystemBackendVolume class connects an IBMTSSVC_Cluster instance with the IBMTSSVC_BackendVolume instances that are visible in the fibre-channel SAN.

References

The IBMTSSVC_SystemBackendVolume class extends the CIM_SystemDevice class and has the references shown in Table 133.

Table 133. IBMTSSVC_SystemBackendVolume references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Cluster	Min, Max, Aggregate	The Cluster
Part Component	IBMTSSVC_Backend Volume	Weak	The BackendVolume

IBMTSSVC_SystemCandidateVolume

The IBMTSSVC_SystemCandidateVolume class associates a Cluster or RemoteCluster and its Candidate Volumes.

References

The IBMTSSVC_SystemCandidateVolume class extends the CIM_SystemDevice class and has the references shown in Table 134.

Table 134. IBMTSSVC_SystemCandidateVolume references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Remote Cluster	Aggregate, Max, Min	The aggregating cluster
Part Component	IBMTSSVC_Candidate Volume	Weak	The Candidate Volume

IBMTSSVC_SystemController

The IBMTSSVC_SystemController class associates an I/O group with the corresponding Controller instances.

References

The IBMTSSVC_SystemController class extends the CIM_SystemDevice class and has the references shown in Table 135.

Table 135. IBMTSSVC_SystemController references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_IOGroup	Min, Max, Aggregate	The I/O group
Part Component	IBMTSSVC_Controller	Weak	The Controller

IBMTSSVC_SystemFCPort

The IBMTSSVC_SystemFCPort class associates a node to its fibre-channel ports.

References

The IBMTSSVC_SystemFCPort class extends the CIM_SystemDevice class and has the references shown in Table 136.

Table 136. IBMTSSVC_SystemFCPort references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Node	Min, Max, Aggregate	The node
Part Component	IBMTSSVC_FCPort	Weak	The fibre-channel port

IBMTSSVC_SystemFeatures

The IBMTSSVC_SystemFeatures class associates the Cluster with its features.

References

The IBMTSSVC_SystemFeatures class extends the CIM_ElementCapabilities class and has the references shown in Table 137.

Table 137. IBMTSSVC_SystemFeatures references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Cluster	Min, Max	The Cluster
Capabilities	IBMTSSVC_Features	Weak	The Features

IBMTSSVC_SystemVolume

The IBMTSSVC_SystemVolume class associates a StorageVolume with the RedundancyGroup to which it is assigned.

References

The IBMTSSVC_SystemVolume class extends the CIM_SystemDevice class and has the references shown in Table 138 on page 262.

Table 138. IBMTSSVC_SystemVolume references

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_IOGroup	Min, Max, Aggregate	The assigned RedundancyGroup
Part Component	IBMTSSVC_Storage Volume	Weak	The StorageVolume

IBMTSSVC_SystemVPD

The IBMTSSVC_SystemVPD class associates the Node with its vital product data (VPD).

References

The IBMTSSVC_SystemVPD class extends the CIM_ElementSettingData class and has the references shown in Table 139.

Table 139. IBMTSSVC_SystemVPD references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Node	Key	The Node
SettingData	IBMTSSVC_NodeVPD	Key	The VPD

Properties

The IBMTSSVC_SystemVPD class has the properties shown in Table 140.

Table 140. IBMTSSVC_SystemVPD properties

Property	Type	Qualifier	Description
IsCurrent	Uint16		An enumerated integer that indicates that the referenced setting is currently being used in the operation of the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Current **2** Is Not Current
IsDefault	Uint16		An enumerated integer that indicates that the referenced setting is a default setting for the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Default **2** Is Not Default

IBMTSSVC_UseOfMessageLog

ManagedSystemElements may record their event, error, or informational data in MessageLogs. The use of a log to hold ManagedSystemElement data is described by this association. The type of data captured by the log can be specified using the RecordedData string property.

References

The IBMTSSVC_UseOfMessageLog class extends the CIM_UseOfMessageLog class and has the references shown in Table 141.

Table 141. IBMTSSVC_UseOfMessageLog references

Name	Reference	Qualifier	Description
Antecedent	IBMTSSVC_MessageLog		The MessageLog
Dependent	IBMTSSVC_Cluster		The ManagedSystem Element for which data is recorded in the MessageLog

Properties

The IBMTSSVC_UseOfMessageLog class has the properties shown in Table 142.

Table 142. IBMTSSVC_UseOfMessageLog properties

Property	Type	Qualifier	Description
RecordedData	String		The description of the use of the Log by the ManagedSystem Element

IBMTSSVC_VolumeSettingData

The IBMTSSVC_VolumeSettingData class extends the CIM_ElementSettingData class.

References

The IBMTSSVC_VolumeSettingData class has the references shown in Table 143.

Table 143. IBMTSSVC_VolumeSettingData references

Name	Reference	Qualifier	Description
ManagedElement	IBMTSSVC_Storage Volume		The managed element
SettingData	IBMTSSVC_Storage Setting		The SettingData object that is associated with the element

Properties

The IBMTSSVC_VolumeSettingData class has the properties shown in Table 144.

Table 144. IBMTSSVC_VolumeSettingData properties

Property	Type	Qualifier	Description
IsCurrent	Uint16		An enumerated integer that indicates that the referenced setting is currently being used in the operation of the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Current **2** Is Not Current
IsDefault	Uint16		An enumerated integer that indicates that the referenced setting is a default setting for the element or that the setting is unknown. **Code** **Semantics** **0** Unknown **1** Is Default **2** Is Not Default

IBMTSSVC_AccountManagementServiceForSystem

The IBMTSSVC_AccountManagementServiceForSystem class connects an IBMTSSVC_Cluster instance with an IBMTSSVC_AccountManagementService.

References

The IBMTSSVC_AccountManagementServiceForSystem class extends the CIM_ManagesAccountOnSystem class and has the references shown in Table 145.

Table 145. IBMTSSVC_AccountManagementServiceForSystem references

Name	Target	Qualifier	Description
Antecedent	IBMTSSVC_Account ManagementService	ReadRole, Min, Max	Represents the SecurityService that provides services for the system
Dependent	IBMTSSVC_Object Manager	ReadRole, Weak	Represents the system that is dependent on the security service

IBMTS_AccountOnCIMOM

The IBMTS_AccountOnCIMOM class connects an IBMTS_Account instance with the IBMTS_ObjectManager instance.

References

The IBMTS_AccountOnCIMOM class extends the CIM_AccountOnSystem class and has the references shown in Table 146.

Table 146. IBMTS_AccountOnCIMOM references

Name	Reference	Qualifier	Description
Group Component	IBMTS_ObjectManager	Aggregate, ReadRole, Min, Max	Represents the aggregation of the CIMOM for the Account
Part Component	IBMTS_Account	ReadRole, Weak	Represents the Account

Properties

The IBMTS_AccountOnCIMOM class has the properties shown in Table 147.

Table 147. IBMTS_AccountOnCIMOM properties

Property	Type	Qualifier	Description
Role	String	ReadRole(Administrator)	Specifies the role of the Account on the CIMOM

IBMTS_AccountOnSystem

The IBMTS_AccountOnSystem class connects an IBMTS_Account instance with an IBMTSSVC_Host instance.

References

The IBMTS_AccountOnSystem class extends the CIM_AccountOnSystem class and has the references shown in Table 148.

Table 148. IBMTS_AccountOnSystem

Name	Reference	Qualifier	Description
Group Component	IBMTSSVC_Cluster	Aggregate, ReadRole, Min, Max	Represents the aggregating system to which access is granted for the Account
Part Component	IBMTS_Account	ReadRole, Weak	Represents the subordinate Account

Properties

The IBMTS_AccountOnSystem class has the properties shown in Table 149 on page 266.

Table 149. IBMTS_AccountOnSystem properties

Property	Type	Qualifier	Description
Role	String	ReadRole(Administrator)	Specifies the role of the Account on the System

IBMTS_CommMechanismForManager

The IBMTS_CommMechanismForManager is an association between an ObjectManager and an ObjectManagerCommunicationMechanism class.

References

The IBMTS_CommMechanismForManager class has the references shown in Table 150.

Table 150. IBMTS_CommMechanismForManager

Name	Target	Qualifier	Description
Antecedent	IBMTS_ObjectManager	Min, Max	Represents the specific ObjectManager whose communication mechanism is described
Dependent	IBMTS_CIMXML Communication	Min	Represents the encoding, protocol, or set of operations that may be used to communicate with the referenced ObjectManager

IBMTS_ElementConformsToProfile

The IBMTS_ElementConformsToProfile connects the ObjectManager and the Server profile.

References

The IBMTS_ElementConformsToProfile class has the references shown in Table 151.

Table 151. IBMTS_ElementConformsToProfile

Name	Target	Qualifier	Description
Conformant Standard	IBMTS_RegisteredProfile		Represents the RegisteredProfile to which the ManagedElement conforms
ManagedElement	IBMTS_ObjectManager		Represents the ManagedElement that conforms to the RegisteredProfile

IBMTS_HostedAccessPoint

The IBMTS_HostedAccessPoint connects the CIM_System and the CIMXMLMechanism profile.

References

The IBMTS_HostedAccessPoint class has the references shown in Table 152.

Table 152. IBMTS_HostedAccessPoint

Name	Target	Qualifier	Description
Antecedent	IBMTS_System	Min, Max	Represents the hosting system
Dependent	IBMTS_CIMXML Communication	Weak	Represents the SAP(s) that are hosted on this system

IBMTS_HostedService

The IBMTS_HostedService is an association between a Service and the System on which the functionality resides. A System may host many Services. Services are weak with respect to their hosting System. A Service is hosted on the System where the LogicalDevices or SoftwareFeatures that implement the Service are located.

References

The model does not represent Services hosted across multiple systems. This is modeled as an ApplicationSystem that acts as an aggregation point for Services, that are each located on a single host.

The IBMTS_HostedService class has the references shown in Table 153.

Table 153. IBMTS_HostedService

Name	Target	Qualifier	Description
Antecedent	IBMTS_System	Min, Max	Represents the hosting system
Dependent	IBMTS_Object Manager	Weak	Represents the service hosted on this system

IBMTS_IndicationFiltersConformsToProfile

The IBMTS_IndicationFiltersConformsToProfile connects the IndicationFilter and the Server profile.

References

The IBMTS_IndicationFiltersConformsToProfile class has the references shown in Table 154.

Table 154. IBMTS_IndicationFiltersConformsToProfile

Name	Target	Qualifier	Description
Conformant Standard	IBMTS_RegisteredProfile		Represents the RegisteredProfile to which the ManagedElement conforms

Table 154. IBMTS_IndicationFiltersConformsToProfile (continued)

Name	Target	Qualifier	Description
ManagedElement	IBMTS_IndicationFilter		Represents the IndicationFilter that conforms to the RegisteredProfile

IBMTS_ManagesAccount

The IBMTS_ManagesAccount class connects an IBMTS_AccountManagementService instance with an IBMTS_Account instance.

References

The IBMTS_ManagesAccount class extends the CIM_ManagesAccount class and has the references shown in Table 155.

Table 155. IBMTS_ManagesAccount references

Name	Reference	Qualifier	Description
Antecedent	IBMTS_Account ManagementService	ReadRole	The AccountManagement Service
Dependent	IBMTS_Account	ReadRole	The HardwareAccount

IBMTS_NamespaceInManager

The IBMTS_NamespaceInManager.

References

The IBMTS_NamespaceInManager class has the references shown in Table 156.

Table 156. IBMTS_NamespaceInManager

Name	Target	Qualifier	Description
Antecedent	IBMTS_Object Manager	Min, Max	Represents the ObjectManager containing a Namespace
Dependent	IBMTS_Name Space	Weak	Represents the Namespace in an ObjectManager

Chapter 7. CIM Agent methods

This chapter describes the intrinsic and extrinsic methods that the CIM Agent classes provide.

These methods are required for implementing the functionality of the CIM Agent.

Intrinsic methods

Intrinsic methods are provided for modeling a typical CIM operation.

Originating from the CIM and WBEM standards, intrinsic methods are provided for modeling a typical CIM operation. Intrinsic methods provide the basic means that enable you to work with an object model.

The CIM Agent for the SAN Volume Controller supports the intrinsic methods shown in Table 157.

Table 157. Supported intrinsic methods

Functional group	Method name
Association traversal	Associators()
	AssociatorNames()
	References()
	ReferenceNames()
Basic read	EnumerateClasses()
	EnumerateClassNames()
	EnumerateInstance()
	EnumerateInstanceNames()
	GetClass()
	GetInstance()
	GetProperty()
Basic write	SetProperty()
Instance manipulation	DeleteInstance()
	CreateInstance()
	ModifyInstance()
Query execution	ExecQuery()

Associators()

The Associators() method is used to enumerate the classes or instances that are associated with a CIM object.

Parameters

Table 158 on page 270 shows the parameters you can specify for the Associators() method.

© Copyright IBM Corp. 2003, 2004

Table 158. Associators() parameters

Parameter	Type	Description
ObjectName	COP*	Defines the class name or instance name that is the source of the association
AssocClass	String	If not NULL, indicates that all objects must be associated to the source object through an instance of this class or one of its subclasses.
ResultClass	String	If not NULL, indicates that all returned objects must be instances of this class or one of its subclasses or be this class.
Role	String	If not NULL, indicates that each return object must be associated to the source object through an association in which the source object plays the specified role. The name of the property in the association class that refers to the source object must match the value of this parameter.
ResultRole	String	If not NULL, indicates that each returned object must be associated to the source object through an association in which the returned object plays the specified role. That is, the name of the property in the association class that refers to the returned object must match the value of this parameter.
IncludeQualifiers	Boolean	TRUE returns all qualifiers for the class, its properties, methods, or method parameters. FALSE returns no qualifiers.
IncludeClassOrigin	Boolean	TRUE returns the CLASSORIGIN attribute of the class.
* CIMObjectPath		

Return values

The Associators() method enumerates the specified classes or instances or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_FAILED

AssociatorNames()

The AssociatorNames() method is used to enumerate the names of the classes or instances that are associated with a CIM object.

Parameters

Table 159 on page 271 shows the parameters you can specify for the AssociatorNames() method.

Table 159. AssociatorNames() parameters

Parameter	Type	Description
ObjectName	COP	Defines the class name or instances name that is the source of the association.
AssocClass	String	If not NULL, indicates that all object paths returned identify an object that is associated to the source object through an instance of this class or one of its subclasses.
ResultClass	String	If not NULL, indicates that all returned object paths must identify instances of this class or one of its subclasses or must be this class.
Role	String	If not NULL, the name of the property in the association class that refers to the source object must match the value of this parameter.
ResultRole	String	If not NULL, the name of the property in the association class that refers to the returned object must match the value of this parameter.

Return values

The AssociatorNames() method enumerates the names of the classes or instances or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_FAILED

CreateInstance()

The CreateInstance() method is used to create a new instance of an object in the target namespace.

Parameters

The new instance must be based on a class that is already defined in the namespace.

Table 160 shows the parameters you can specify for CreateInstance() method.

Table 160. CreateInstance() parameters

Parameter	Type	Description
Instance	String	Defines the name of the instance to create

Return values

The CreateInstance() method creates the specified class or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER

- CIM_ERR_INVALID_CLASS
- CIM_ERR_NOT_FOUND
- CIM_ERR_FAILED

DeleteInstance()

The DeleteInstance() method is used to remove a single instance of an object from the target namespace.

Parameters

Table 161 shows the parameters you can specify for the DeleteInstance() method.

Table 161. DeleteInstance() parameters

Parameter	Type	Description
InstanceName	String	Defines the name of the instance to delete

Return values

The DeleteInstance() method deletes the specified instance or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_NOT_FOUND
- CIM_ERR_FAILED

EnumerateClasses()

The EnumerateClasses() method is used to enlist all subclasses of a single object class or all classes of the same object type in the target namespace.

Parameters

Table 162 shows the parameters you can specify for the EnumerateClasses() method.

Table 162. EnumerateClasses() parameters

Parameter	Type	Description
ClassName	String	Defines the name of the class for which subclasses are to be returned. If this parameter is NULL, all base classes within the target namespace are returned.
DeepInheritance	Boolean	TRUE returns all subclasses of the specified class. FALSE returns only immediate child subclasses.
LocalOnly	Boolean	TRUE returns all properties, methods, and qualifiers, that are overridden within the definition of the class.
IncludeQualifiers	Boolean	If set to TRUE, returns all qualifiers for the class, its properties, methods, or method parameters; if set to FALSE, returns no qualifiers

Table 162. EnumerateClasses() parameters (continued)

Parameter	Type	Description
IncludeClassOrigin	Boolean	If set to TRUE, returns the CLASSORIGIN attribute of the class

Return values

The EnumerateClasses() method enumerates the specified one or more classes or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_FAILED

EnumerateClassNames()

The EnumerateClassNames() method is used to enlist the names of all subclasses of a single object class or the names of all classes of the same object type in the target namespace.

Parameters

Table 163 shows the parameters you can specify for the EnumerateClassNames() method.

Table 163. EnumerateClassNames() parameters

Parameter	Type	Description
ClassName	String	Defines the name of the class for which subclasses are to be returned. If this parameter is NULL, all base classes within the target namespace are returned.
DeepInheritance	Boolean	TRUE returns all subclasses of the specified class. FALSE returns only immediate child subclasses.

Return values

The EnumerateClassNames() method enumerates the specified name of one or more classes or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_FAILED

EnumerateInstances()

The EnumerateInstances() method is used to enlist all instances of the same object class in the target namespace.

Parameters

Table 164 shows the parameters you can specify for the EnumerateInstances() method:

Table 164. EnumerateInstances() parameters

Parameter	Type	Description
ClassName	String	Defines the name of the class for which instances are to be returned.
DeepInheritance	Boolean	TRUE returns all instances and all properties of the instance, including those added by creating subclasses. FALSE returns only properties defined for the specified class.
LocalOnly	Boolean	TRUE returns all properties, methods, and qualifiers that are overridden within the definition of the class.
IncludeQualifiers	Boolean	TRUE returns all qualifiers for each instance, its properties, methods, or method parameters. FALSE returns no qualifiers
IncludeClassOrigin	Boolean	TRUE returns the CLASSORIGIN attribute of the class within the instance.

Return values

The EnumerateInstances() method enumerates the specified instances or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_FAILED

EnumerateInstanceNames()

The EnumerateInstanceNames() method is used to enlist all the names of the instances of the same object class in the target namespace.

Parameters

Table 165 shows the parameters you can specify the following parameters of the EnumerateInstanceNames() method.

Table 165. EnumerateInstanceNames() parameters

Parameter	Type	Description
ClassName	String	Defines the name of the class for which instances are to be returned.

Return values

The EnumerateInstanceNames() method enumerates the specified names of the instances or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED

- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_FAILED

ExecQuery()

The ExecQuery() method is used to execute a query against the target namespace.

Parameters

Table 166 shows the parameters you can specify for the ExecQuery() method.

Table 166. ExecQuery() parameters

Parameter	Type	Description
QueryLanguage	String	Defines the query language in which the query parameter is expressed.
Query	String	Defines the query to be executed.

Return values

The ExecQuery() method retrieves one or more classes or instances or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_FAILED

GetClass()

The GetClass() method is used to retrieve a single object class from the target namespace.

Parameters

Table 167 shows the parameters you can specify for the GetClass() method.

Table 167. GetClass() parameters

Parameter	Type	Description
ClassName	String	Defines the name of the class to retrieve
LocalOnly	Boolean	If set to TRUE, returns all properties, methods, and qualifiers overridden within the definition of the class
IncludeQualifiers	Boolean	If set to TRUE, returns all qualifiers for the class, its properties, methods, or method parameters; if set to FALSE, returns no qualifiers
IncludeClassOrigin	Boolean	If set to TRUE, returns the CLASSORIGIN attribute of the class

Return values

The GetClass() method returns the specified class or one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_FAILED

GetInstance()

The GetInstance() method is used to retrieve a single instance of an object from the target namespace.

Parameters

Table 168 shows the parameters you can specify for the GetInstance() method.

Table 168. GetInstance() parameters

Parameter	Type	Description
InstanceName	String	Defines the name of the instance to retrieve
LocalOnly	Boolean	If set to TRUE, returns all properties, methods, and qualifiers overridden within the definition of the class
IncludeQualifiers	Boolean	If set to TRUE, returns all qualifiers for the class, its properties, methods, or method parameters; if set to FALSE, returns no qualifiers
IncludeClassOrigin	Boolean	If set to TRUE, returns the CLASSORIGIN attribute of the class

Return values

The GetInstance() method returns the specified class or one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_NOT_FOUND
- CIM_ERR_FAILED

GetProperty()

The GetProperty() method is used to retrieve the whole instance and return one specific property from this instance.

Parameters

Table 169 on page 277 shows the parameters you can specify for the GetProperty() method.

Table 169. GetProperty() parameters

Parameter	Type	Description
InstanceName	String	Defines the name of the instance
Property	String	The name of the property whose value is to be returned from the instance

Return values

The GetProperty() method returns the specified property of the target instance or one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_NOT_FOUND
- CIM_ERR_NO_SUCH_PROPERTY
- CIM_ERR_FAILED

ModifyInstance()

The ModifyInstance() method is used to modify an existing instance of an object in the target namespace.

Parameters

Table 170 shows the parameters you can specify for the ModifyInstance() method.

Table 170. ModifyInstance() parameters

Parameter	Type	Description
InstanceName	String	Defines the name of the instance to modify

Return values

The ModifyInstance() method modifies the specified instance or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_NOT_FOUND
- CIM_ERR_FAILED

References()

The References() method is used to enumerate the association objects that refer to a particular target class or instance.

Parameters

Table 171 shows the parameters you can specify for the References() method.

Table 171. References() parameters

Parameter	Type	Description
ObjectName	String	Defines the class name or instance name whose referring objects are to be returned.
ResultClass	String	If not NULL, indicates that all returned objects must be instances of this class or one of its subclasses or must be this class.
Role	String	If not NULL, must be a valid property name. Each returned object must refer to the target object through a property whose name matches the value of this parameter.
IncludeQualifiers	Boolean	TRUE returns all qualifiers for the class, its properties, methods, or method parameters. FALSE returns no qualifiers.
IncludeClassOrigin	Boolean	TRUE returns the CLASSORIGIN attribute of the class.

Return values

The References() method enumerates the association objects or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_FAILED

ReferenceNames()

The ReferenceNames() method is used to enumerate the association objects that refer to a particular target class or instance.

Parameters

Table 172 shows the parameters you can specify for the ReferenceNames() method.

Table 172. ReferenceNames() parameters

Parameter	Type	Description
ObjectName	String	Defines the class name or instance name whose referring objects are to be returned.
ResultClass	String	If not NULL, indicates that all returned object paths must be object paths of instances of this class or one of its subclasses, or must be this class.
Role	String	If not NULL, must be a valid property name. Each returned object must refer to the target object through a property whose name matches the value of this parameter.

Return values

The ReferenceNames() method enumerates the association objects or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_NOT_FOUND
- CIM_ERR_NO_SUCH_PROPERTY
- CIM_ERR_FAILED

SetProperty()

The SetProperty() method is used to define a single property value of an instance in the target namespace.

Parameters

Table 173 shows the parameters you can specify for the SetProperty() method.

Table 173. SetProperty() parameters

Parameter	Type	Description
InstanceName	String	Defines the name of the instance
PropertyName	String	The name of the property whose value is to be defined.

Return values

The SetProperty() method defines the property name of the target instance or returns one of the following error codes:

- CIM_ERR_ACCESS_DENIED
- CIM_ERR_INVALID_NAMESPACE
- CIM_ERR_INVALID_PARAMETER
- CIM_ERR_INVALID_CLASS
- CIM_ERR_NOT_FOUND
- CIM_ERR_NO_SUCH_PROPERTY
- CIM_ERR_TYPE_MISMATCH
- CIM_ERR_FAILED

Extrinsic methods

Extrinsic methods are specific to CIM object classes and are defined by the object model provider according to a specific SMI-S schema. The extrinsic methods add functionality to the CIM object classes.

The CIM Agent for the SAN Volume Controller supports the extrinsic methods listed in Table 174 on page 280.

Table 174. Supported extrinsic methods

Class	Method name
IBMTSSVC_BackendVolume	GetFreeExtents()
IBMTSSVC_Chassis	IsCompatible()
IBMTSSVC_ClusteringService	AddNode()
	BackupConfiguration()
	Clean()
	DeleteConfigurationBackups()
	Dump()
	EvictNode()
	GetDump()
	GetResetPasswordChangeFeatureStatus()
	ListConfigurationBackups()
	ModifyIPAddress()
	ModifyResetPasswordChangeFeature()
	RestoreConfiguration()
	SetLocale()
	SetTimeZone()
	SetPasswords()
	Shutdown()
	StartService()
	StartStatisticsCollection()
	StopService()
	StopStatisticsCollection()
IBMTSSVC_ControllerConfigurationService	AttachDevice()
	CreateProtocolControllerWithPorts()
	DeleteProtocolController()
	DetachDevice()
IBMTSSVC_Job	KillJob()
IBMTSSVC_MessageLog	CancelIteration()
	ClearLog()
	DeleteRecord()
	FixRecord()
	GetAllRecords()
	GetRecord()
	ModifyErrorSettings()
	PositionAtRecord()
	PositionToFirstRecord()
	PositionToFirstRecordRoot()
	PositionToFirstRecordType()
	UnfixRecord()
	WriteRecord()

Table 174. Supported extrinsic methods (continued)

Class	Method name
IBMTSSVC_PrimordialStoragePool	GetSupportedSizes()
	GetSupportedSizeRange()
IBMTSSVC_PrivilegeManagementService	AssignAccess()
	RemoveAccess()
IBMTSSVC_Provider	Add2062Cluster()
	Add2145Cluster()
	Create2062Cluster()
	Reload2062Node()
	RemoveCluster()
	Reset2062Node()
IBMTSSVC_ServiceModeService	Clean()
	Dump()
	Enter()
	Exit()
	GetDump()
	Upgrade()
IBMTSSVC_StorageCapabilities	CreateSetting()
IBMTSSVC_StorageConfigurationService	AttachReplica()
	CreateOrModifyStoragePool()
	CreateOrModifyElementFromStoragePool()
	CreateRemoteClusterPartnership()
	CreateReplica()
	CreateSynchronizedSet()
	DeleteRemoteClusterPartnership()
	DeleteStoragePool()
	DeleteSynchronizedSet()
	IncludeBackendVolume()
	MigrateVolume()
	ModifySynchronization()
	ModifySynchronizedSet()
	ReturnToStoragePool()
	RequestDiscovery()
	SetIOGroup()
	SetQuorum()
	StartService()
	StopService()

Table 174. Supported extrinsic methods (continued)

Class	Method name
IBMTSSVC_StorageHardwareID ManagementService	AddHardwareIDsToCollection()
	CreateHardwareIDCollection()
	CreateStorageHardwareID()
	DeleteHardwareIDCollection()
	DeleteStorageHardwareID()
IBMTSSVC_StoragePool	GetSupportedSizes()
	GetSupportedSizeRange()

Add2062Cluster()

The Add2062Cluster() command can be used to configure the ICAT to work with an existing 2062 cluster.

Parameters

The Add2062Cluster() method belongs to the IBMTSSVC_Provider class. Table 175 shows the parameters you can specify for the Add2062Cluster() method.

Table 175. Add2062Cluster() parameters

Parameter	Type	Description
Cluster	IBMTSSVC_Cluster REF	A reference to the cluster that was added
ClusterIP	String	The IP of the cluster to be added
ClusterName	String	The name of the cluster to be added
Password	String	The password required to log on to the switch
SwitchIDs	String[]	This parameter identifies all switches that the cluster spans. It must be specified, even if the cluster contains nodes from one switch only.
User	String	The user name required to log on to the switch

Return values

The Add2062Cluster() method returns one of the following error codes:

- 0: The cluster was added successfully.
- 2: The attempt failed.
- 5: The number or type of parameters that have been passed is invalid.
- ..: DMTF Reserved.
- 0x8000: Connection to cluster refused.
- 0x8001: Syntax error in cluster name.
- 0x8002: Invalid node.
- 0x8003: Invalid user name or password.
- 0x8004: Syntax error in switch IP.
- 0x8005: Syntax error in cluster IP.

- 0x8006: Invalid slot.

Add2145Cluster()

The Add2145Cluster() command can be used to configure the ICAT to work with an existing 2145 cluster.

Parameters

The Add2145Cluster() method belongs to the IBMTSSVC_Provider class. Table 176 shows the parameters you can specify for the Add2145Cluster() method.

Table 176. Add2145Cluster() parameters

Parameter	Type	Description
Cluster	IBMTSSVC_Cluster REF	A reference to the cluster that was added
ClusterIP	String	The IP of the cluster to be added

Return values

The Add2145Cluster() method returns one of the following error codes:

- 0: The cluster was added successfully.
- 2: The attempt failed.
- 5: The number or type of parameters that have been passed is invalid.
- ..: DMTF Reserved.
- 0x8000: Connection to cluster refused.
- ..: Vendor Reserved.
- 0x8005: Syntax error in cluster IP.

AddHardwareIDsToCollection()

The AddHardwareIDsToCollection() method adds a StorageHardwareID to a HardwareIDCollection.

Parameters

When a StorageHardwareID is added to a collection, the corresponding host object is deleted on the device and the WWPN is added to the host representing the collection. The AddHardwareIDsToCollection() method belongs to the IBMTSSVC_StorageHardwareIDManagementService class.

Table 177 shows the parameters you can specify for the AddHardwareIDsToCollection() method.

Table 177. AddHardwareIDsToCollection() parameters

Parameter	Type	Description
HardwareIDs	String[]	An array that contains the string representations of COPs of StorageHardwareIDs to be added immediately to the collection. Alternatively, IDs can contain the WWPN. In this case, the creation of StorageHardwareIDs is circumvented.

Table 177. AddHardwareIDsToCollection() parameters (continued)

Parameter	Type	Description
Collection	CIM_SystemSpecific Collection REF	The IBMTSSVC_HardwareId Collection to add the IDs to.

Return values

The AddHardwareIDsToCollection() method returns one of the following error codes:

- 0: The collection was successfully created.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed.
- 5: One of the parameters is invalid.
- 0x1000: The StorageHardwareID could not be found or is already member of another collection.
- 0x1001: Implementation does not support device collections.
- 0x1002: Input devices cannot be used in this collection.
- 0x8100: One or more parameters is out of cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

AddNode()

The AddNode() method is used to add an IBMTSSVC_CandidateNode instance to an IBMTSSVC_Cluster instance.

Parameters

The AddNode() method belongs to the IBMTSSVC_ClusteringService class.

When invoked, the AddNode() method automatically chooses the IBMTSSVC_RedundancyGroup for the candidate node. If the IBMTSSVC_RedundancyGroup instances have only one existing node, the method selects the one whose identifier contains the smallest number. If no such IBMTSSVC_RedundancyGroup exists, the method selects an empty IBMTSSVC_RedundancyGroup whose identifier contains the smallest number.

Table 178 shows the parameters you can specify for the AddNode() method.

Table 178. AddNode() parameters

Parameter	Type	Description
CS	COP	Defines the IBMTSSVC_CandidateNode instance to be added that is in the same cluster as the IBMTSSVC_CLusteringService instance
Set	String	The IO group to add the node to. This must be of type IBMTSSVC_IOGroupSet, belong to the same Cluster as the Service hosting this method and contain zero or one nodes.
Name	String	The name by which the new code will be known by the cluster.

Return values

The AddNode() method returns one of the following error codes:

- 0: The node was successfully added.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: The submitted ComputerSystem was not a IBMTSSVC_CandidateNode
- 0x8001: All redundancy groups already have two nodes assigned.
- 0x8002; The submitted ExtraCapacitySet was not a IBMTSSVC_IGroupSet.
- 0x8003; The submitted IOGroupSet already has two assigned nodes.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

AssignAccess()

The AssignAccess() method is used to create a temporary IBMTSSVC_AccessControlInformation instance that contains the IBMTSSVC_AuthorizationSubject, IBMTSSVC_AuthorizationTarget, IBMTSSVC_HostedACI, and IBMTSSVC_AuthorizedUse associations in the CIMOM repository.

Parameters

The AssignAccess() method belongs to the IBMTSSVC_AuthorizationService class.

Table 179 shows the parameters you can specify for the AssignAccess() method.

Table 179. AssignAccess() parameters

Parameter	Type	Description
Activities	Uint16[]	MUST be NULL unless the Privilege is NULL on input. This parameter specifies the activities to be granted or denied. **Code** **Semantics** **1** Other **2** Create **3** Delete **4** Detect **5** Read **6** Write **7** Execute **..** DMTF Reserved **16000..65535** Vendor Reserved
ActivityQualifiers	String[]	MUST be NULL unless Privilege is NULL on input. Defines the activity qualifiers for the Activities to be granted or denied.
PrivilegeGranted	Boolean	MUST be NULL unless Privilege is NULL on input. Indicates whether the rights defined by the parameters in this call should be granted or denied to the named subject/target pair.

Table 179. AssignAccess() parameters (continued)

Parameter	Type	Description
QualifierFormats	Uint16[]	MUST be NULL unless Privilege is NULL on input. Defines the qualifier formats for the corresponding ActivityQualifiers. **Code** **Semantics** **2** Class Name **3** <Class.>Property **4** <Class.>Method **5** Object Reference **6** Namespace **7** URL **8** Directory/File Name **9** Command Line Instruction **10..15999** DMTF Reserved **16000..65535** Vendor Reserved
Subject	CIM_ManagedElement REF	Defines the IBMTSSVC_HardwareAccount instance that is in the same IBMTSSVC_Cluster as the IBMTSSVC_AuthorizationService instance
Target	CIM_ManagedElement REF	On input, this reference MUST be either NULL or refer to an instance of AuthorizedPrivilege that is used as a template.

Return values

The AssignAccess() method returns one of the following error codes:

- 0: All instances were successfully created.
- 2: An unexpected error occurred.
- 3: Timeout
- 4: Failedx
- 5: The number or type of parameters that have been passed is incorrect.
- 6..15999: DMTF Reserved
- 16000: Unsupported Subject
- 16001: Unsupported Privilege
- 16002: Unsupported Target
- 16003: Authorization Error
- 16004: NULL not supported
- 16005..31999: Method Reserved
- 32000..65535: Vendor Specific

AttachDevice()

The AttachDevice() method is used to attach an IBMTSSVC_StorageVolume instance to an IBMTSSVC_Controller instance that is associated with an IBMTSSVC_AccessControlInformation instance and has the AuthorizationView parameter set to true.

Parameters

The AttachDevice() method belongs to the IBMTSSVC_Controller class.

The provider must verify that unit numbers are unique for each initiator. When the ProtocolController is already part of an AuthorizedTarget association, the provider should update the access configuration in the underlying hardware when AttachDevice is called.

Table 180 shows the parameters you can specify for the AttachDevice() method.

Table 180. AttachDevice() parameters

Parameter	Type	Description
Device	COP	Defines the volume instance to be attached. Must belong to the same RedundancyGroup as the controller and be of type IBMTSSVC_StorageVolume.
[DeviceNumber]		The logical unit number (LUN) at which the volume will be exposed to all hosts connected to this controller.
[Force]	Boolean	When false (the default), attempting to attach a volume that is already attached to another controller will fail.
ProtocolController		The controller to which you attach the volume must belong to the same cluster as this service.

Return values

The AttachDevice() method returns one of the following error codes:

- 0: The volume was successfully attached.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x1000: The device is not a volume of the controller's RedundancyGroup.
- 0x1001: The specified device number is already occupied.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

AttachReplica()

AttachReplica() creates a copy relationship between two volumes.

Parameters

Table 181 shows the parameters you can specify for the AttachReplica() method.

Table 181. AttachReplica() parameters

Parameter	Type	Description
[BackgroundCopyRate]	Uint16	Specifies the priority of the background copy rate "0–100". Scale is not in percent "non-linear".

Table 181. AttachReplica() parameters (continued)

Parameter	Type	Description
CopyType	String	Determines the type of the copy relationship. In CIM this is Sync, Async or UnSynchAssoc. In SAN Volume Controller terminology, this translates to “flash” or “remote.” For a CopyType=3 (Sync) a remote copy is created, for CopyType=4 (UnSyncAssoc) a flash copy is created.
[ElementName]	String	The name of the IBMTSSVC_StorageSynchronized association.
[Set]	String	Defines the IBMTSSVC_SynchronizedSet
SourceElement		The source volume. Required to be an IBMTSSVC_StorageVolume.
[Synchronized]	Boolean	Valid for CopyType “Sync” only. If true, the SAN Volume Controller assumes that both source and target already contain identical data and no initial synchronization has to be performed.
TargetElement		The target volume. Can be an IBMTSSVC_StorageVolume or an IBMTSSVC_CandidateVolume.

Return values

The AttachReplica() method returns one of the following error codes:

- 0: The copy relationship was established successfully.
- 4: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: CopyType is neither 2 nor 3.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

BackupConfiguration()

The BackupConfiguration() command is used to execute the configuration backup script. The script stores the current cluster configuration associated with the current instance of ClusteringService in an XML file.

Parameters

The file is named svc.config.backup.xml and is stored in the directory backup/<clustername> in the CIMOM home directory. If a current cluster back file already exists, the file will be renamed with a .bak extension on the original backup file name. Any current .bak files of the same name will be overwritten. This file is designed for file restoration, if an error or corruption occurs during the backup process. If the .bak files are used for restore, they must be renamed manually.

Table 182 on page 289 shows the parameters you can specify for the BackupConfiguration() method.

Table 182. BackupConfiguration() parameters

Parameter	Type	Description
Force	Boolean	True/False statement. If true, command continuation is forced. False is the default.
FilePath	String	The backup file path.
Messages	String[]	The errors/warnings received from the backup script.

Return values

The BackupConfiguration() method returns one of the following error codes:

- 0: The backup was successful.
- 2: An unexpected error occurred and the command failed.
- 5: The number or type of passed parameters is incorrect.
- 0x8001: The backup script returned an error.
- 0x8002: The backup file download via scp, failed.
- 0x8003: The backup directory could not be created.
- 0x8004: The old backup file could not be renamed or deleted.

CancelIteration()

The CancelIteration() method requests that an iteration of the Log, identified by the IterationIdentifier input parameter, be stopped. CancelIteration() is a method that belongs to the IBMTSSVC_MessageLog class.

Parameters

Table 183shows the parameters you can specify for the CancelIteration() method.

Table 183. CancelItration() parameters

Parameter	Type	Description
IterationIdentifier	String	The current iterator.

Return values

The CancelIteration() method returns one of the following error codes.

- 0: The method completed successfully.
- 2: Unknown.
- 3: Timeout.
- 4: Failed.
- 5. The number or type of parameters that have been passed is incorrect.

Clean()

The Clean() command is used to clean the dump directories on the node.

Parameters

The Clean() method belongs to the IBMTSSVC_ServiceModeService class. Table 184 on page 290 shows the parameters you can specify for the Clean()

method,

Table 184. Clean() parameters

Parameter	Type	Description
Filter	String	The filter's syntax. If a directory is specified, with no file filter, all relevant dump/log files in this directory are cleaned. The allowable directory arguments are: dumps (which cleans all files in all subdirectories including), dumps/configs, dumps/elogs, dumps/feature, dumps/iostats, dumps/iotrace and home/admin. In addition to the directory, a file filter can be specified.
SMNode	IBMTSSVC_Node	Specifies the node that the dump file is deleted from. A dump file on the config node will be deleted, if nothing is specified.

Return values

The Clean() method returns one of the following error codes:

- 0: The Clean() method was successful.
- 2: An unexpected error occurred and the command failed.
- 5: The wrong number or type of parameter has been passed.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

ClearLog()

The ClearLog() method deletes all entries in the error log.

Return values

If the request is not supported, check the Capabilities array that a value of 6 (″Clear Log Supported″) is specified. In a subclass, the set of possible return codes could be described using a ValueMap qualifier on the method.The ClearLog() method belongs to the IBMTSSVC_MessageLog class.

The ClearLog() method returns one of the following error codes:

- 0: All entries in the error log were deleted.
- 2: An unexpected error occurred.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

Create2062Cluster()

The Create2062Cluster() command can be used to create a Cluster.

Parameters

After the cluster has been successfully created, the configuration of the ICAT will be updated with the new cluster information. The Create2062Cluster() method belongs to the IBMTSSVC_Provider class. Table 185 on page 291 shows the parameters

you can specify for the Create2062Cluster() method.

Table 185. Create2062Cluster() parameters

Parameter	Type	Description
ClusterIP	String	The desired IP of the cluster to be created
ClusterName	String	The desired name of the cluster to be created
Node	Uint8	The node on the 2062 blade where the cluster will be created
Password	String	The password required to log on to the switch
Slot	Uint8	The slot of the 2062 blade where the cluster will be created
SwitchIDs	String[]	This parameter identifies all switches that the cluster spans. It must be specified, even if the cluster contains nodes from one switch only.
SwitchIP	String	The IP of the switch where the 2062 blades will reside
User	String	The user name required to log on to the switch

Return values

The Create2062Cluster() method returns one of the following error codes:

- 0: The cluster was established successfully.
- 2: The attempt failed.
- 5: The number or type of parameters that have been passed is invalid.
- ..: DMTF Reserved.
- 0x8000: Connection to cluster refused.
- 0x8001: Syntax error in cluster name.
- 0x8002: Invalid node.
- 0x8003: Invalid user name or password.
- 0x8004: Syntax error in switch IP.
- 0x8005: Syntax error in cluster IP.
- 0x8006: invalid slot.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

CreateHardwareIDCollection()

The CreateHardwareIDCollection method creates an instance of IBMTSSVC_HardwareIdCollection, which is represented on the device by a host object.

Parameters

The HardwareIdCollection aggregates StorageHardwareIDs. If a StorageHardwareID is added to a collection, then the corresponding host object is deleted on the device and the WWPN is added to the host that represents the

collection. The CreateHardwareIDCollection() method belongs to the IBMTSSVC_ControllerConfigurationService class.

Table 186 shows the parameters you can specify for the CreateHardwareIDCollection() method.

Table 186. CreateHardwareIDCollection() parameters

Parameter	Type	Description
ElementName	String	The name of the collection.
HardwareIDs	String[]	An array that contains the string representations of COPs of StorageHardwareIDs to be added immediately to the collection. Alternatively, IDs can contain the WWPN. In this case, the creation of StorageHardwareIDs is circumvented.
Collection	CIM_System SpecificCollection REF	The IBMTSSVC_HardwareId Collection to add the IDs to.

Return values

The CreateHardwareIDCollection() method returns one of the following error codes:

- 0: The collection was successfully created.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed.
- 5: One of the parameters is invalid.
- 0x1000: The StorageHardwareID could not be found or is already member of another collection.
- 0x1001: Implementation does not support hardware ID collections.
- 0x1002: Input hardware IDs cannot be used in same collection.
- 0x8100: One or more parameters is out of cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

CreateOrModifyStoragePool()

The CreateOrModifyStoragePool() method is used to create or modify an IBMTSSVC_StoragePool.

Parameters

You can use the CreateOrModifyStoragePool() method to create an IBMTSSVC_StoragePool if the Pool parameter is set to null, or you can modify an existing IBMTSSVC_StoragePool if the Pool parameter is not null. The CreateOrModifyStoragePool() method belongs to the IBMTSSVC_StorageConfigurationService class.

Table 187 on page 293 shows the parameters you can specify for the CreateOrModifyStoragePool() method.

Table 187. CreateOrModifyStoragePool() parameters

Parameter	Type	Pool creation description	Pool modification description
BlockSize	Uint16	The block size (also known as extent size) for the new Pool. Supported values are 16, 32, 64, 128, 256, or 512 MB. The default value is 16 MB. The block size defines the granularity by which the capacity is managed by SVC. For example, if you select a block size of 256 MB, all StorageVolumes will occupy space in multiples of 256 MB. Thus, a 300 MB StorageVolume will allocate 512 MB of Pool capacity.	Must be null. The block size can be set at creation time only.
ElementName	String	The ElementName of the Pool to be created. If null, the system will assign a name.	The new name for the Pool.
[Extent]	String	Points to the StorageExtent	Points to the StorageBackend Volume
Force	Boolean	Not used.	If set to True, deletion of managed disks (MDisks) is forced. Force is ignored if disks are only are added.
Goal	CIM_StorageSetting REF	Contains the desired name and extent size for the new pool. If Goal is different from the static instance in the persistence layer, the method will fail.	The desired name and extent size for the new pool. If Goal is different from the static instance in the persistence layer, the method will fail.
InExtents	String[]	The BackendVolumes from which to build the Pool. Mutually exclusive with InPools. If InExtents is given, Size will be ignored.	StorageBackend Volumes, which translate to managed disks (MDisks) in SVC terminology. These are the volumes that will be added to the pool or removed from the pool. They must belong to the same cluster as the pool and must not be aggregated by any other pool. If Size is less than the actual size of the Pool, BackendVolumes passed in here will be removed.

Table 187. CreateOrModifyStoragePool() parameters (continued)

Parameter	Type	Pool creation description	Pool modification description
InPools	String[]	The PrimordialPool from which to take the BackendVolumes. Mutually exclusive with InExtents.	The PrimordialPool from which to take additional BackendVolumes. Mutally exclusive with InExtents.
Job	CIM_ConcreteJob REF	Set to Null.	Set to Null.
[Pool]	String	Set to Null.	Specifies the StoragePool that is in the same cluster as the Storage ConfigurationService instance.
Pool	CIM_StoragePool REF	The name of the new pool.	The parameter is left unchanged by the method and therefore contains the passed value from invocation.
[Size]		The desired size of the new Pool. When InPools is given, it will take as many BackendVolumes as necessary from the PrimordialPool in order to achieve the requested size. If InExtents is given, this parameter will be ignored. The size of the new storage pool is given by the sum of the sizes of the aggregated extents.	The desired new size of the Pool. When InPools is given, it will take BackendVolumes from the PrimordialPool in order to achieve the requested size. If InExtents is given, this parameter will be ignored. See InExtents for more information. The size of the new storage pool is given by the sum of the sizes of the aggregated extents. You cannot shrink a Pool using this method and passing InPools.
Size	Uint64	The real allocated size of the Pool in bytes.	The real allocated size of the Pool in bytes.

Return values

The CreateOrModifyStoragePool() method returns one of the following error codes:

- 0: The pool was successfully created.
- 2: An unknown error occurred.
- 4: An unexpected error occurred.

- 3: The action timed out.
- 5: At least one of the parameters was invalid.
- 6: In use.
- 4096: Method parameters checked. Job started.
- 4097: Size not supported.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

CreateOrModifyElementFromStoragePool()

The CreateOrModifyElementFromStoragePool() method is used to create an IBMTSSVC_StorageVolume instance if TheElement parameter is null, or you can modify an IBMTSSVC_StorageVolume instance if TheElement parameter is not null.

Parameters

The CreateOrModifyElementFromStoragePool() method belongs to the IBMTSSVC_StorageConfigurationService class.

Table 188 shows the parameters you can specify for the CreateOrModifyElementFromStoragePool() method.

Table 188. CreateOrModifyElementFromStoragePool() parameters

Parameter	Type	Description	
		Volume creation	**Volume modification**
Backend Volumes	String[]	An array that contains the BackendVolumes that will store the data from the volume. This parameter is mandatory if the virtualization type is ″sequential″ or ″image″. For all VirtualizationTypes exept ″image,″ all BackendVolumes must belong to the StoragePool in InPool. If the virtualization type is ″image,″ then the BackendVolume must belong to a PrimordialStoragePool.	In the case of Volume expansion, a list of BackendVolumes on which the additional capacity will be allocated can be submitted. In all other cases, this parameter must be null.
Element Name	String	The ElementName of the volume to be created. If null, the system will assign a name.	The new name for the Volume.
ElementType	Uint16	ElementType=2, IBMTSSVC_StorageVolume.	ElementType=2, IBMTSSVC_StorageVolume.
Format	Boolean	Specifies whether or not the volume will be formatted on creation. Default is false.	Whether or not the additional volume capacity will formatted on expansion. Default is false.
Goal	CIM_Managed Element REF	Contains special settings for the new volume. Must be a valid StorageSetting for Volumes if submitted.	Must be a valid StorageSetting for Volumes, if submitted.

Table 188. CreateOrModifyElementFromStoragePool() parameters (continued)

Parameter	Type	Description	
		Volume creation	**Volume modification**
InPool	CIM_Storage Pool REF	The IBMTSSVC_StoragePool from which the volume is to be allocated. The pool and the StorageConfigurationService must belong to the same cluster. Must be set if TheElement is null.	Set to null.
IOGroup	IBMTSSVC_ IOGroup REF	The IOGroup that the StorageVolume will be assigned to. If null, the method will choose the IOGroup with the fewest number of virtual disks (VDisks) assigned.	Must be null.
Job	CIM_Concrete Job REF	Set to null.	Set to null.
Preferred Node	IBMTSSVC_ Node REF	The preferred Node for Volume access. If IOGroup is null, this parameter must also be null. Otherwise, it must belong to the IOGroup specified.	
Size	Uint64	The size of the volume in bytes. This parameter directly corresponds to the size CLI parameter. SVC supports only sizes that are multiples of 512 bytes. If the size doesn't fulfill this criteria, then the method fails with ″size not supported″ and the next largest multiple of 512 bytes in size.	The size of the volume in bytes. The CLI amount parameter is calculated as follows: amount = Size - current size. If amount is positive, then the volume will be expanded. If the amount is negative, it will be reduced. SVC supports only sizes that are multiples of 512 bytes. If the size doesn't fulfill this criteria, then the method fails with ″size not supported″ and the next largest multiple of 512 bytes in size.
TheElement	CIM_Logical Element REF	Specifies the IBMTSSVC_StorageVolume instance to be created. If null, a new StorageVolume will be allocated from InPool. If not null the passed StorageVolume will be expanded or reduced, depending on the Size parameter.	Specifies the IBMTSSVC_StorageVolume instance to be modified (reduced or expanded).

Table 188. CreateOrModifyElementFromStoragePool() parameters (continued)

Parameter	Type	Description	
		Volume creation	**Volume modification**
Virtualization Type	Uint8	Sets the type of Volume created. May be ″striped″ (0), ″sequential″ (1) or ″image″ (2). Default is ″striped.″	Must be null.

Return values

The CreateOrModifyElementFromStoragePool() method returns one of the following error codes:

- 0: The volume was successfully created.
- 4: An unexpected error occurred.
- 5: At least one of the parameters was invalid.
- 0x1001: The requested size is not a multiple of 512. The nearest supported size greater than the requested size is returned in Size.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

CreateProtocolControllerWithPorts()

The CreateProtocolControllerWithPorts() method creates an IBMTSSVC_Controller.

Parameters

The controller can be used to attach Volumes using AttachDevice(). The controller is created in the Common Information Model Object Manager (CIMOM) repository. A controller is bound to an IOGroup and can therefore contain only FCPorts of this IOGroup. The CreateProtocolControllerWithPorts() method belongs to the IBMTSSVC_ControllerConfigurationService class.

Table 189 shows the parameters you can specify for the CreateProtocolControllerWithPorts() method.

Table 189. CreateProtocolControllerWithPorts() parameters

Parameter	Type	Description
ElementName	String	The name is automatically assigned and cannot be chosen individually. Therefore, this parameter is required to be null.
Identity	CIM_ManagedElement REF	The IBMTSSVC_HardwareIdCollection or IBMTSSVC_StorageHardwareID to which the Volumes attached to the ProtocolController shall be exported.
Ports	String[]	The list of ports that will be associated with the controller. All ports must belong to the same IOGroup. The created controller will contain all FCPorts of an IOGroup even if a subset is submitted here.
Protocol	Uint16	Is required to be 2.

Table 189. CreateProtocolControllerWithPorts() parameters (continued)

Parameter	Type	Description
ProtocolController	CIM_ProtocolController REF	The created IBMTSSVC_Controller is returned here.
Privilege	CIM_Privilege REF	If not null, required to be the default static Privileges instance from the persistence layer.

Return values

The CreateProtocolControllerWithPorts() method returns one of the following error codes:

- 0: A clone was successfully created.
- 2: An unexpected error occurred.
- 5: The wrong number or type of parameters were passed.
- 0x8000: All ports are required to belong to the same IOGroup.
- 0x8100: COPs of a Cluster (which is different to this service's cluster) where submitted.

CreateRemoteClusterPartnership()

The CreateRemoteClusterPartnership() method is used to establish a one-way partnership between an IBMTSSVC_Cluster instance and an IBMTSSVC_CandidateCluster instance.

Parameters

The method must be executed on both the source cluster and the candidate cluster to establish a fully functional synchronous copy partnership. The CreateRemoteClusterPartnership() method belongs to the IBMTSSVC_StorageConfigurationService class.

Table 190 shows the parameters you can specify for the CreateRemoteClusterPartnership() method.

Table 190. CreateRemoteClusterPartnership() parameters

Parameter	Type	Description
RemoteCluster	IBMTSSVC_ Candidate Cluster REF	The cluster that a relationship will be established to. Cluster membership checks are required.
[Bandwidth]	Uint16	The bandwidth for the copy operation in megabytes (MB).

Return values

The CreateRemoteClusterPartnership() method returns one of the following error codes:

- 0: The cluster partnership was successfully established.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed.

- 5: The wrong number or type of parameters were passed.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

CreateReplica()

The CreateReplica() method is used to create a replica of the source volume for a copy relationship.

Parameters

The CreateReplica() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 191 shows the parameters you can specify for the CreateReplica() method.

Table 191. CreateReplica() parameters

Parameter	Type	Description
CopyType	Uint16	The type of copy relationship. Possible values are 3 (remote copy) or 4 (FlashCopy). Remote copies can be established on the same cluster only.
[ElementName]	String	The name of the replica to be created. If null, then the system will assign a name.
Job	CIM_ConcreteJob REF	The object used to monitor and terminate the copy process.
SourceElement	CIM_LogicalElement REF	The source StorageVolume for the replica.
TargetElement	CIM_LogicalElement REF	The target StorageVolume for the replica.
TargetSettingGoal	CIM_StorageSetting REF	The StorageSetting object to be matched by the replica. Passed to CreateOrModifyElement FromStoragePool.
TargetPool	String	The IBMTSSVC_StoragePool that will be used for the target volume.

Return values

The CreateReplica() method returns one of the following error codes:

- 0: The cluster partnership was successfully established.
- 4: An unexpected error occurred.
- 5: The number or type of parameters that were passed is incorrect.
- 0x8000: A CopyType other than 3 or 4 was used.
- 0x8100: One or more of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

CreateSetting()

The CreateSetting() method is used to create and populate a StorageSetting instance from a StorageCapability instance.

Parameters

This removes the need to populate default settings and other settings in the context of each StorageCapabilities (which could be numerous).

The CreateSetting() method belongs to the IBMTSSVC_StorageCapabilities class. Table 192 shows the parameters you can specify for the CreateSetting() method.

Table 192. CreateSetting() parameters

Parameter	Type	Description
NewSetting	CIM_StorageSetting REF	Reference to the created StorageSetting instance
SettingType	Uint16	The value of this parameter is without meaning to the SAN Volume Controller. For both Default and Goal, the same StorageSetting is returned.

Return values

The CreateSetting() method returns one of the following error codes.

- 0: The method completed successfully.
- 2: Unspecified error.
- 3: Timeout.
- 4: The method failed.
- 5. The number or type of parameters that have been passed is incorrect.
- ..: DMTF reserved.
- 32768..65535: Vendor specific.

CreateStorageHardwareID()

The CreateStorageHardwareID() method creates an instance of IBMTSSVC_StorageHardwareID.

Parameters

On the device level, the instance is represented by a single-port host object with the name prefix ″cimhwid″.

Table 193 shows the parameters you can specify for the CreateStorageHardwareID() method.

Table 193. CreateStorageHardwareID() parameters

Parameter	Type	Description
ElementName	String	The name of the new HardwareID instance. Must be identical to ID.
StorageID	String	The value used by the SecurityService to represent identity. In this case, a PortWWN.
IDType	Uint16	The type of the ID property. In this case, 2 (PortWWN).

Table 193. CreateStorageHardwareID() parameters (continued)

Parameter	Type	Description
OtherIDType	String	The type of the storage ID when the IDType is ″Other.″
Setting	CIM_StorageClient SettingData REF	Required to be null.
HardwareID	CIM_StorageHardwareID REF	The COP of the created StorageHardwareID.

Return values

The CreateStorageHardwareID() method returns one of the following error codes:

- 0: The volume was successfully detached.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x1000: The WWPN is already assigned to an existing StorageHardwareID.
- 0x1001: The IDType is not 2.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

CreateSynchronizedSet()

The CreateSynchronizedSet() method is used to create an IBMTSSVC_SynchronizedSet instance for aggregating IBMTSSVC_StorageSynchronized associations for FlashCopy or remote copy relationships.

Parameters

Some devices may not support SynchronizedSets. In order to find out if SynchronizedSets are supported, call GetSupportedSetTypes. The CreateSynchronizedSet() method belongs to the IBMTSSVC_StorageConfigurationService class.

Table 194 shows the parameters you can specify for the CreateSynchronizedSet() method.

Table 194. CreateSynchronizedSet() parameters

Parameter	Type	Description
CopyType	Uint16	The type of copy relationship (3 for remote copy; 4 for FlashCopy).
[ElementName]	String	The name of the copy relationship.
[RemoteCluster]	IBMTSSVC_ Remote Cluster REF	The remote cluster for the ConsistencySet. Only StorageSynchronized with volumes on this remote cluster can be added to this set. This setting is valid only for CopyType 3 (remote copy). The default is the local cluster. This setting must be null if the CopyType is 4 (FlashCopy).

Table 194. CreateSynchronizedSet() parameters (continued)

Parameter	Type	Description
[Set]	CIM_Synchronized Set REF	The IBMTSSVC_FlashCopySynchronized Set or IBMTSSVC_SyncCopySynchronized Set instance that is created.

Return values

The CreateSynchronizedSet() method returns one of the following error codes:

- 0: The SynchronizedSet was created successfully.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed.
- 5: The number or type of parameters that were passed is incorrect.
- 6: A copy type other than 3 or 4 was used.
- 8: SynchronizedSets are not supported.
- 0x8100: One or more parameters were out of cluster scope.
- 0x9000 to 0x9FFF: This range represents SAN Volume Controller return codes and messages.

DeleteConfigurationBackup()

The DeleteConfigurationBackup() command is used to delete the backups in the Backup directory.

Parameters

Table 195 shows the parameters you can specify for the DeleteConfigurationBackup() method.

Table 195. DeleteConfigurationBackup() parameters

Parameter	Type	Description
Backup	String	The name of the backup to delete

Return values

The DeleteConfigurationBackup() method returns one of the following error codes:

- 0: The restore was successful.
- 2: An unexpected error occurred.
- 5: The given backup could not be found.
- 0x8000: The backup directory deletion failed. This could be caused by a sharing violation.

DeleteHardwareIDCollection()

The DeleteHardwareIDCollection() method deletes an SVC host.

Parameters

The DeleteHardwareIDCollection() method belongs to the IBMTSSVC_ControllerConfigurationService class.

Table 196 shows the parameters you can specify for the DeleteHardwareIDCollection() method.

Table 196. DeleteHardwareIDCollection() parameters

Parameter	Type	Description
Collection	CIM_System SpecificCollection REF	The COP of the IBMTSSVC_Host to be deleted.
Force	Boolean	Optionally specifies that the deletion will be forced (if set to true). Otherwise the deletion would fail if a privilege is still associated with the collection. If specified, the host will be deleted even if it is member of a LUN mapping.

Return values

The DeleteHardwareIDCollection() method returns one of the following error codes:

- 0: The volume was successfully detached.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x1000: The device has no ProtocolControllerForUnit association to this controller.
- 0x8000: Collection is associated with a privilege, and the Force parameter was not specified.
- 0x8100: One or more parameters were out of cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

DeleteProtocolController()

The DeleteProtocolController() method deletes a controller from the Common Information Model Object Manager (CIMOM) repository or from SAN Volume Controller, respectively.

Parameters

The DeleteProtocolController() method belongs to the IBMTSSVC_ControllerConfigurationService class. Table 197 shows the parameters you can specify for the DeleteProtocolController() method.

Table 197. DeleteProtocolController() parameters

Parameter	Type	Description
DeleteLogicalUnits	Boolean	If true, all StorageVolumes which are exclusively attached to the submitted controller are deleted as well. The default is false.

Table 197. DeleteProtocolController() parameters (continued)

Parameter	Type	Description
ProtocolController	CIM_ProtocolController REF	The controller to be deleted.

Return values

The DeleteProtocolController() method returns one of the following error codes:

- 0: The controller was successfully deleted.
- 2: An unexpected error occurred.
- 5: The wrong number or type of parameters were passed.
- 0x1000: At least one of the attached StorageVolumes is attached to another controller and has therefore not been deleted.
- 0x8100: The passed controller and the service owning the method belong to different clusters.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

DeleteRecord()

The DeleteRecord() method belongs to the IBMTSSVC_MessageLog class.

Parameters

Table 198shows the parameters you can specify for the DeleteRecord() method.

Table 198. DeleteRecord() parameters

Parameter	Type	Description
IterationIdentifier	String	
PositionToNext	Boolean	When set to TRUE, requests the IterationIdentifier to be advanced to the next record, after the current entry is deleted. If set to FALSE, IterationIdentifier is set to the previous record.
RecordNumber	Uint64	
RecordData	Uint8[]	

Return values

The DeleteRecord() method returns one of the following error codes:

- 0: The method completed successfully.
- 2: Unknown.
- 3: Timeout.
- 4: Failed.
- 5. The number or type of parameters that have been passed is incorrect.

DeleteRemoteClusterPartnership()

The DeleteRemoteClusterPartnership() method is used to remove a partnership between two IBMTSSVC_Cluster instances.

Parameters

You must execute the method on both clusters to delete a fully functional synchronous copy partnership. The DeleteRemoteClusterPartnership() method belongs to the IBMTSSVC_StorageConfigurationService class.

Table 199 shows the parameters you can specify for the DeleteRemoteClusterPartnership() method.

Table 199. DeleteRemoteClusterPartnership() parameters

Parameter	Type	Description
RemoteCluster	IBMTSSVC_Remote Cluster REF	The name of the candidate remote cluster. Cluster membership checks are required.

Return values

The DeleteRemoteClusterPartnership() method returns one of the following error codes:

- 0: The cluster partnership was successfully deleted.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed
- 5: The wrong number or type of parameters was passed.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

DeleteStorageHardwareID()

The DeleteStorageHardwareID() method deletes a StorageHardwareID.

Parameters

It removes the associations and aggregations, including CIM_ConcreteDependency and CIM_AuthorizedSubject. The DeleteStorageHardwareID() method belongs to the IBMTSSVC_ControllerConfigurationService class.

Table 200 shows the parameters you can specify for the DeleteStorageHardwareID() method.

Table 200. DeleteStorageHardwareID() parameters

Parameter	Type	Description
HardwareID	CIM_StorageHardwareID REF	The IBMTSSVC_StorageHardwareID to delete.

Table 200. DeleteStorageHardwareID() parameters (continued)

Parameter	Type	Description
Force	Boolean	Optionally specifies that the deletion will be forced (if set to true). The StorageHardwareID will be deleted even if it is associated with a Privilege. If specified, the ID will be deleted even if any active LUN masking is assigned to it.

Return values

The DeleteStorageHardwareID() method returns one of the following error codes:

- 0: The volume was successfully detached.
- 2: An unexpected error occurred.
- 3: The action timed out.
- 4: The action failed.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x1000: The StorageHardwareID could not be found.
- 0x8000: The HardwareAccount is still bound to AuthorizationSubject.
- 0x8100: One or more parameters were out of cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

DeleteSynchronizedSet()

The DeleteSynchronizedSet() method is used to delete a SynchronizedSet if it does not contain any StorageSynchronized associations.

Parameters

The DeleteSynchronizedSet() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 201 shows the parameters you can specify for the DeleteSynchronizedSet() method.

Table 201. DeleteSynchronizedSet()

Parameter	Type	Description
Force	Boolean	If set to False, the delete will fail if any StorageSynchronized are member of the set. The default is false. When Force is set to True, all contained StorageSynchronized will be moved out of the Set before it will be deleted. Therefore, the StorageSynchronized survive as stand-alone copy mappings.
Set	CIM_Synch-ronizedSet REF	The SynchronizedSet to be deleted. The InstanceID corresponds to the SVC consistency_grp ID.

Return values

The DeleteSynchronizedSet() method returns one of the following error codes:

- 0: The SynchronizedSet was successfully deleted.
- 2: An unknown error occurred.

- 3: The action timed out.
- 4: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 6: There are still StorageSynchronized associations in the set when it should be empty. Any StorageSynchronized associations must be removed before deletion of the set can be executed, or the Force flag must be set.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

DeleteStoragePool()

The DeleteStoragePool() method is used to delete an IBMTSSVC_StoragePool instance if it does not contain any IBMTSSVC_StorageVolume instances.

Parameters

The DeleteStoragePool() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 202 shows the parameters you can specify for the DeleteStoragePool() method.

Table 202. DeleteStoragePool()

Parameter	Type	Description
[Force]	Boolean	When set to True, the Pool will be deleted despite constraints (for example, no StorageVolume or BackendVolume contained). The default is False.
Job	CIM_Concrete Job REF	Set to null.
Pool	CIM_Storage Pool REF	The IBMTSSVC_StoragePool to be deleted. Contains the mdisk_grp_ID in Name. The pool and the StorageConfigurationService must belong to the same cluster.

Return values

The DeleteStoragePool() method returns one of the following error codes:

- 0: The pool was successfully deleted.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed
- 6: The method is in use.
- 5: The number or type of parameters that have been passed is incorrect.
- 4096: Method parameters checked. Job started.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

DetachDevice()

The DetachDevice() method detaches a volume from the controller.

Parameters

The DetachDevice() method belongs to the IBMTSSVC_ControllerConfigurationService class. Table 203 shows the parameters you can specify for the DetachDevice() method.

Table 203. DetachDevice() parameters

Parameter	Type	Description
Device	CIM_LogicalDevice REF	The volume to be detached. There must be an IBMTSSVC_SCSILUN association between this volume and the controller.
ProtocolController	CIM_ProtocolController REF	The controller from which to detach the volume.

Return values

The DetachDevice() method returns one of the following error codes:

- 0: The volume was successfully detached.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x1000: The device has no ProtocolControllerForUnit association to this controller.
- 0x8100: One or more parameters were out of cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

Dump()

The Dump() method is used to dump the contents of error log, the config log, or the feature log to a text file.

Parameters

The Dump() method belongs to the IBMTSSVC_ServiceModeService class. Table 204 shows the parameters you can specify for the Dump() method.

Table 204. Dump() parameters

Parameter	Type	Description
Type	Uint16	Decides which dump type will be generated.
FileNamePrefix	String	The dump is directed to a file with a system defined name, if this is not supplied. If supplied, a filename is created from the prefix and a timestamp. It takes the form of <FileNamePrefix> _NN_YYMMDD_HHMMSS. NN is the current configuration log id. In case of feature logs, this parameter must be null.
GeneratedFile	String	The generated file name.
SMNode	IBMTSSVC_ServiceMode Node REF	The node that is in service mode where dump files will be created.

Return values

The Dump() method returns one of the following error codes:

- 0: The text file dump was successful.
- 2: An unexpected error occurred and the command failed.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: A type greater than 2 was passed.
- 0x8001: A file prefix was passed at the same time as the feature log type.
- 0x9000: to 0x9FFF: San Volume Controller return codes and messages.

Enter()

The Enter() method puts a node in service mode.

Parameters

After this happens, the ServiceModeService methods can be executed against the returned service mode node and the objects prefixed with IBMTSSVC_SM objects can beretrieved for this node. If the config node is in service mode, no other commands for this cluster will be available.

The Enter() method belongs to the IBMTSSVC_ServiceModeService class. Table 205 shows the parameters you can specify for the Enter() method.

Table 205. Enter() parameters

Parameter	Type	Description
Node	IBMTSSVC_Node REF	Specifies the node that shall be put in service mode
SMNode	IBMTSSVC_Service ModeNode REF	The reference to the node that is now in service mode

Return values

The Enter() method returns one of the following error codes:

- 0: The Enter() method was successful.
- 5: The wrong number or type of parameter has been passed.
- ..: DMTF reserved.
- 0x8100: Cluster scope violation.

EvictNode()

The EvictNode() method is used to remove an IBMTSSVC_Node instance from an IBMTSSVC_Cluster instance.

Parameters

The EvictNode() method belongs to the IBMTSSVC_ClusteringService class. Table 206 on page 310 shows the parameters you can specify for the EvictNode() method.

Table 206. EvictNode() parameters

Parameter	Type	Description
CS	COP	Defines the IBMTSSVC_Node instance to be added that is in the same cluster as the IBMTSSVC_ClusteringService instance

Return values

The EvictNode() method returns one of the following error codes:

- 0: The node was successfully added.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: The submitted computer system was not an IBMTSSVC_CandidateNode.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

Exit()

The Exit() method will exit the service mode for a specific node and restart it in normal operating mode.

Parameters

The Exit() method belongs to the IBMTSSVC_ServiceModeService class. Table 207 shows the parameters you can specify for the Exit() method.

Table 207. Exit() parameters

Parameter	Type	Description
SMNode	IBMTSSVC_Service ModeNode REF	Specifies the node that will be restarted in normal mode.

Return values

The Exit() method returns one of the following error codes:

- 0: The Exit() method was successful.
- 5: The wrong number or type of parameter has been passed.
- ..: DMTF reserved.
- 0x8100: Cluster scope violation.
- 0x9000 to 0x9FFF: Device error codes.

FixRecord()

The FixRecord() method fixes one entry in the log (marks the entry as fixed).

Parameters

The FixRecord() method belongs to the IBMTSSVC_MessageLog class. Table 208 on page 311 shows the parameters you can specify for the FixRecord() method.

Table 208. FixRecord() parameters

Parameter	Type	Description
RecordNumber	Uint64	Passed to the CLI command in the -d parameter.

Return values

The FixRecord() method returns one of the following error codes:

- 0: The method completed successfully.
- 2: An unexpected error occurred.
- 5: One of the mandatory parameters is missing.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

GetAllRecords()

The GetAllRecords() method retrieves the list of logs from the cluster. The GetAllRecords() method belongs to the IBMTSSVC_MessageLog class.

Parameters

Table 209 shows the parameters you can specify for the GetAllRecords() method.

Table 209. GetAllRecords() parameters

Parameter	Type	Description
ErrorOnly	Boolean	If set to TRUE or null, only error records are returned. If FALSE, all log entries are returned.
Records	String[]	The file as a string array.

Return values

The GetAllRecords() method returns one of the following error codes:

- 0: The method completed successfully.
- 2: An unexpected error occurred.
- 0x8000: Records not found.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

GetDump()

The GetDump() method is used to retrieve a log file. The feature log will be automatically decoded.

Parameters

The GetDump() method belongs to the IBMTSSVC_ServiceModeService class. Table 210 on page 312 shows the parameters you can specify for the GetDump() method.

Table 210. GetDump() parameters

Parameter	Type	Description
FilePath	String	The fully specified file name. File names can be viewed in the IBMTSSVC_Dump instance. Allowed paths are: dumps/configs, dumps/elogs, dumps/feature, dumps/iostats, dumps/iotrace and home/admin.
SMNode	IBMTSSVC_Node REF	Specified node from for retrieving the dump. The config node dump will be returned if nothing is specified.
File	String[]	The file as a string array.

Return values

The GetDump() method returns one of the following error codes:

- 0: The GetDump() method was successful.
- 5: The number or type of parameters that have been passed is incorrect.
- ..: DMTF reserved.
- 0x8000: The connection to the cluster has been lost, or failed to connect to a node.
- x8001: A given file path was not found for CISCO..
- 0x8100: One or more parameters where out of the cluster scope.
- 0x9000..0x9FFF: Device error codes.

GetFreeExtents()

GetFreeExtents() returns the number of free extents on a BackendVolume.

Parameters

These extents are the blocks which the SAN Volume Controller uses for capacity management. They do not correspond to the CIM_StorageExtent class.

The GetFreeExtents() method belongs to the IBMTSSVC_BackendVolume class. Table 211 shows the parameters you can specify for the GetFreeExtents() method.

Table 211. GetFreeExtents() parameters

Parameter	Type	Description
FreeExents		The number of free extents on this BackendVolume.

Return values

The GetFreeExtents() method returns one of the following error codes:

- 0: Method successfully completed.
- 2: An unexpected error occurred.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

GetRecord()

The GetRecord() method retrieves the list of logs from the cluster, returns the log entry specified by the IterationIdentifier, and increments the IterationIdentifier by one position if PositionToNext set to TRUE.

Parameters

The GetRecord() method belongs to the IBMTSSVC_MessageLog class. Table 212 shows the parameters you can specify for the GetRecord() method.

Table 212. GetRecord() parameters

Parameter	Type	Description
[IterationIdentifier]	String	The pointer to the record to be retrieved. The tokens are evaluated and the corresponding command is called.
IterationIdentifier	String	The new IterationIdentifier. The SequenceNumber was modified according to the parameters that were passed in. If PositionToNext was true and there is no next entry, null will be returned here.
PositionToNext	String	If this value is true, the IterationIdentifier is advanced by one position. Otherwise the old IterationIdentifier will be returned.
RecordData	Uint8[]	The entry in byte representation (UTF-8). Use String.getBytes(UTF-8).
RecordNumber	Uint64	This value equals the SequenceNumber token of the IterationIdentifier that is returned.

Return values

The GetRecord() method returns one of the following error codes:

- 0: The method completed successfully.
- 2: An unexpected error occurred.

GetResetPasswordChangeFeatureStatus()

The GetResetPasswordChangeFeatureStatus() method is used to retrieve the current status of the rest password change feature.

Parameters

The GetResetPasswordChangeFeatureStatus() method belongs to the IBMTSSVC_ClusteringService class. Table 213 on page 314 shows the parameters you can specify for the GetResetPasswordChangeFeatureStatus() method.

Table 213. GetResetPasswordChangeFeatureStatus() properties

Parameter	Type	Description
Enable	Boolean	If set to True, indicates that the password reset feature is enabled; if set to False, indicates that the feature is disabled

Return values

The GetResetPasswordChangeFeatureStatus() method returns one of the following error codes:

- 0: The feature status has been successfully retrieved.
- 2: An unexpected error occurred.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

GetSupportedSizeRange()

The GetSupportedSizeRange() method is used to query the volume sizes that are supported.

Parameters

The GetSupportedSizeRange() method belongs to the IBMTSSVC_StoragePool class. Table 214 shows the parameters you can specify for the GetSupportedSizeRange() method.

Table 214. GetSupportedSizeRange() parameters

Parameter	Type	Description
ElementType	Uint16	The type of element for which supported size ranges are reported. **Code** **Semantics** **2** Storage Pool **3** Storage Volume
[Goal]	COP	Specifies the size requirements.
Minimum VolumeSize	Uint64	Specifies the minimum size (MB) to query
Maximum VolumeSize	Uint64	Specifies the maximum size (MB) to query
VolumeSize Divisor	Uint64	A volume/pool size must be a multiple of this value

Return values

The GetSupportedSizeRange() method returns one of the following error codes:

- 0: Parameters are valid.
- 2: Use GetSupportedSizes instead.
- 5: The number or type of parameters that have been passed is incorrect.

GetSupportedSizes()

You can use the GetSupportedSizes() method to query the supported volume sizes.

Parameters

The GetSupportedSizes() method is derived from the IBMTSSVC_StoragePool class. Table 215shows the parameters of the GetSupportedSizes() method.

Table 215. GetSupported Sizes() parameters

Parameter	Type	Description
ElementType	Uint16	The type of element for which supported sizes are reported. **Code** **Semantics** **2** Storage Pool **3** Storage Volume
Goal	CIM_StorageSetting REF	Specifies the size requirements
Sizes	uint64[]	List of support sizes for a Volume/Pool creation or modification (MB)

Return values

The GetSupportedSizes() method returns one of the following error codes:

- 1: Method not supported.
- 2: Use the GetSupportedSizeRange method instead.

IncludeBackendVolume()

The IncludeBackendVolume() method is used to reinstate an IBMTSSVC_BackendVolume instance that was ejected by an IBMTSSVC_Cluster.

Parameters

The IncludeBackendVolume() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 216 shows the parameters you can specify for the IncludeBackendVolume() method.

Table 216. IncludeBackendVolume() parameters

Parameter	Type	Description
Volume	IBMTSSVC_Backend Volume REF	The IBMTSSVC_BackendVolume instance to be reinstated. Must belong to the same cluster as the StorageConfigurationService.

Return values

The IncludeBackendVolume() method returns one of the following error codes:

- 0: The volume was successfully included.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: The volume had not been ejected.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

ListConfigurationBackups()

The ListConfigurationBackups() command lists the backups in the Backup directory. Only the directory names are reported.

Parameters

Table 217 shows the parameters you can specify for the ListConfigurationBackups() method.

Table 217. ListConfigurationBackups() parameters

Parameter	Type	Description
Backup	String	Each array element contains the name of one backup available in the backu/directory.

Return values

The ListConfigurationBackups() method returns the following error code:

- 0: The backup was successful.

MigrateVolume()

The MigrateVolume() method is used to migrate an IBMTSSVC_StorageVolume instance to another IBMTSSVC_StoragePool instance.

Parameters

The MigrateVolume() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 218 shows the parameters you can specify for the MigrateVolume() method.

Table 218. MigrateVolume() parameters

Parameter	Type	Description
Job	CIM_ConcreteJob REF	The object that can be used to monitor the migration progress.
NumberOfThreads	Uint8	Tthe number of copy threads used for the migration. Can be 1 to 4.
TargetPool	IBMTSSVC_Storage Pool REF	The IBMTSSVC_StoragePool instance to which the IBMTSSVC_StorageVolume instance will be migrated. Must be different from the pool the volume is currently a member of.
Volume	IBMTSSVC_Storage Volume REF	The IBMTSSVC_StorageVolume to be migrated. The complete data of this volume will be copied to the new location.

Return values

The MigrateVolume() method returns one of the following error codes:

- 0: The volume was successfully migrated.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8100: One of the parameters was out of the cluster scope.

- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

ModifyErrorSettings()

The ModifyErrorSettings() method allows you to specify what happens when an error or event is logged to the error log. The ModifyErrorSettings() method belongs to the IBMTSSVC_MessageLog class.

Parameters

Table 219 shows the parameters you can specify for the ModifyErrorSettings() method.

Table 219. ModifyErrorSettings() parameters

Parameter	Type	Description
[EmailAddress]	String	The email address of which to send email notification.
[EmailAlert]	String	The email setting (when to raise an email notification). • all = raise email for all errors logged • hardware_only = raise email for errors but not object state changes • none = do not raise email for any errors (default cluster setting)
[SNMP Community]	String	The SNMP community string.
[SNMP ManagerIP]	String	The IP address of the host system running the SNMP manager software.
[SNMPTrap]	String	The SNMP trap setting (when to raise a trap). Allowed values: • all = raise SNMP for all errors logged • no_state = raise SNMP for errors but not object state changes • none = do not raise SNMP for any errors (default cluster setting)

Return values

The ModifyErrorSettings() method returns the following error codes:

- 0: The method successfully specified action.
- 2: A command failed.
- 0x8200: The method executed successfully but one or more parameters were ignored.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

ModifyIPAddress()

The ModifyIPaddress() command is used to modify the IP address in the cluster, change the entry in the provider-config.xml and reload the configuration.

Parameters

Table 220 shows the parameters you can specify for the Modifyipaddress() method.

Table 220. Modifyipaddress() parameters

Parameter	Type	Description
ClusterIP	String	Specifies and validates the new cluster IP address.

Return values

The Modifyipaddress() method returns one of the following error codes:

- 0: The Modifyipaddress command was successful.
- 2: An unexpected error occurred and the command failed.
- 5: One of the mandatory parameters is missing or invalid.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

ModifyResetPasswordChangeFeature()

The ModifyResetPasswordChangeFeature() method is used to enable or disable the password reset feature.

Parameters

The ModifyResetPasswordChangeFeature() method belongs to the IBMTSSVC_ClusteringService class. Table 221 shows the parameters you can specify for the ModifyResetPasswordChangeFeature() method.

Table 221. ModifyResetPasswordChangeFeature() parameters

Parameter	Type	Description
Enable	Boolean	If set to True, enables the password reset feature; if set to False, disables the feature

Return values

The ModifyResetPasswordChangeFeature() method returns one of the following error codes:

- 0: The passwords were changed.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

ModifySynchronization()

The ModifySynchronization() method is used to modify a FlashCopy or synchronous copy relationship between two IBMTSSVC_StorageVolume instances based on the specified type of operation.

Parameters

The ModifySynchronization() method belongs to the IBMTSSVC_StorageConfigurationService class.

Table 222 shows the parameters you can specify for the ModifySynchronization() method.

Table 222. ModifySynchronization() parameters

Parameter	Type	Description
Operation	Uint16	These commands can be executed only if the copy mapping does not belong to a consistency group (check SynchronizedSetID in StorageSynchronized). **Code** **Semantics** **2** Detach. Delete the copy mapping. **3** Fracture. Suspend the synchronization between two storage objects. The association and changes are remembered to allow a fast resynchronization. This can be used during a backup cycle to allow one of the objects to be copied while the other remains in production. **4** Resync Replica. Re-establish the synchronization of a replica. If CopyJob is Sync or Async, this will negate the action of a previous fracture operation. **5** Restore from Replica. Renew the contents of the original storage object from a replica. **6** Prepare. Prepare the participating volumes for a point-in-time copy. **7** Unprepare. **8** Quiesce. **9** Unquiesce. **10** Reset To Sync. **11** Reset To Async. **0x8000** Switch. Switch primary relationship. **0x8001** Stop.
[AllowAccess]	Boolean	If set to True, the target will be accessible for I/O after fracture of a sync copy. The default is False. The parameter is ignored for all operations except sync copy fracture.
[Clean]	Boolean	If set to True, the target is assumed to be clean (initialized with zeros), thus no initialize will be done before resync of a sync copy. The default is False. Ignored for all other operations except resync of a sync copy.

Table 222. ModifySynchronization() parameters (continued)

Parameter	Type	Description
[Direction]	Boolean	If set to True, the master (SyncedSystemElement) will become the source of the sync copy relationship. If set to False, the auxiliary (SyncedElement) will become the source of the sync copy relationship. By default, the direction will be autonomically chosen (reverse on switch and keep on resync). Valid for switch and resync operation of sync copy only. Ignored in all other cases.
[Force]	Boolean	If set to True, the operation will be forced. The default is False. Applies only on resync and detach of FlashCopy mappings and resync of sync copy mappings. Ignored in all other cases.
Job	CIM_ConcreteJob REF	The object that is used to monitor and terminate the copy process.
Synchronization	CIM_Storage Synchronized REF	The copy relationship to be modified: IBMTSSVC_FlashCopyStorage Synchronized or IBMTSSVC_SyncCopyStorage Synchronized.

Return values

The ModifySynchronization() method returns one of the following error codes:

- 0: The FlashCopy mapping was successfully established.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed
- 5: The number or type of parameters that have been passed is incorrect.
- 6: The method is in use.
- 0x1000: The CLI copy command ran and a job object was returned.
- 0x8001: The operation that was submitted is not allowed in the current state of the StorageSycnhronized. For example, a prepare operation on a StorageSynchronized is in synchronized state.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

ModifySynchronizedSet()

The ModifySynchronizedSet() method is used to modify a SynchronizedSet based on the specified operation type.

Parameters

The ModifySynchronizedSet() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 223 on page 321 shows the parameters you can specify for the ModifySynchronizedSet() method.

Table 223. ModifySynchronizedSet() parameters

Parameter	Type	Description
Operation	Uint16	**Code** **Semantics** **0** Add. Add a synchronization to the set. **1** Remove. Remove a synchronization from the set. **2** Detach all. Delete all synchronizations in the set. **3** Fracture replicas. Suspend the synchronization between two storage objects. The association and changes are remembered to allow a fast resynchronization. This can be used during a backup cycle to allow one of the objects to be copied while the other remains in production. **4** Resync replicas. Re-establish the synchronizations of all replicas in the set. If CopyJob is Sync or Async, this will negate the action of a previous fracture operation. **5** Restore from replica. Renew the contents of the original storage objects from the replicas. **6** Prepare all. Prepare the participating volumes for a point-in-time copy. **7** Unprepare all. **8** Quiesce replicas. **9** Unquiesce replicas. **0x8000** Switch. Switch primary relationships. **0x8001** Stop All. Stop copy mapping.
[AllowAccess]	Boolean	If set to True, the target will be accessible for I/O after fracture of a sync copy. The default is False. The parameter is ignored for all operations except sync copy fracture.
[Clean]	Boolean	If set to True, the target is assumed to be clean (initialized with zeros), thus no initialize will be done before resync of a sync copy. The default is False. Ignored for all other operations except resync of a sync copy.

Table 223. ModifySynchronizedSet() parameters (continued)

Parameter	Type	Description
[Direction]	Boolean	If set to True, the master (SyncedSystemElement) will become the source of the sync copy relationship. If set to False, the auxiliary (SyncedElement) will become the source of the sync copy relationship. By default, the direction will be autonomically chosen (reverse on switch and keep on resync). Valid for switch and resync operation of sync copy only. Ignored in all other cases.
[Force]	Boolean	If set to True, the operation will be forced. The default is False. Applies only on resync and detach of FlashCopy mappings and resync of sync copy mappings. Ignored in all other cases.
Job	CIM_ConcreteJob REF	Defines the object used to monitor and terminate the copy process
[Synchronization]	CIM_Storage Synchronized REF	The copy mapping to be added to or removed from the set. Ignore for all other operations.
SynchronizedSet	CIM_Synchronized Set REF	The SynchronizedSet to be modified.

Return values

The ModifySynchronizedSet() method returns one of the following error codes:

- 0: The SyncCopySynchronizedSet was successfully created.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: The action failed.
- 5: The number or type of parameters that have been passed is incorrect.
- 6: The operation is not supported.
- 7: StorageSynchronized not in the set.
- 8: StorageSynchronized already in a set.
- 9: A StorageSynchronized should be added to the set that is not compatible to the set. For example, a FlashCopy synchronized to a sync sopy set.
- 0x1000: Method parameters checked. Job started.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x8001: An operation should be performed which is not allowed on the current SyncState of the set.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

PositionAtRecord()

The PositionAtRecord() method sets the SequenceNumber and the RelPos token of the returned IterationIdentifier, depending on the parameters that were passed in.

Parameters

The PositionAtRecord() method belongs to the IBMTSSVC_MessageLog class. Table 224 shows the parameters you can specify for the PositionAtRecord() method.

Table 224. PositionAtRecord() parameters

Parameter	Type	Description
IterationIdentifier	String	The new IterationIdentifier. The SequenceNumber and the RelPos token were modified according to the parameters that were passed in.
[IterationIdentifier]	String	The current IterationIdentifier (created by a previous position or method) or null.
MoveAbsolute	Boolean	Set to TRUE if the IterationIdentifier should be positioned absolutely. In this case the SequenceNumber is set to the value passed in RecordNumber.
RecordNumber	Uint64	If MoveAbsolute == FALSE: A (signed) offset to the current position in the log. If MoveAbsolute == TRUE: The absolute desired position in the log is the sequence number of the entry. It is not allowed to specify a negative value in RecordNumber. The IterationIdentifier's SequenceNumber is set to this value. If the RelPos token was set in the IterationIdentifier that was passed in it is removed from the IterationIdentifier.

Return values

The PositionAtRecord() method returns the following error code:

- 0: The IterationIdentifier was successfully created.
- 5: One of the mandatory parameters is missing or RecordNumber is negative when not allowed.

PositionToFirstRecord()

The PositionToFirstRecord() method creates an IterationIdentifier that points to the first entry in the log.

Parameters

The IterationIdentifier is used in subsequent GetRecord or Position... calls.

The PositionToFirstRecord() method belongs to the IBMTSSVC_MessageLog class. Table 225 shows the parameters you can specify for the PositionToFirstRecord() method.

Table 225. PositionToFirstRecord() parameters

Parameter	Type	Description
IterationIdentifier	String	The IterationIdentifier which points to the first record.

Return values

The PositionToFirstRecord() method returns the following error code:

- 0: The IterationIdentifier was successfully created.

PositionToFirstRecordRoot()

The PositionToFirstRecordRoot() method creates an IterationIdentifier whose SequenceNumber is not set and whose RootCause token is set to the value that is passed in.

Parameters

The IterationIdentifier is used in subsequent GetRecord or PositionAtRecord calls.

The PositionToFirstRecordRoot() method belongs to the IBMTSSVC_MessageLog class.Table 226 shows the parameters you can specify for the PositionToFirstRecordRoot() method.

Table 226. PositionToFirstRecordRoot() parameters

Parameter	Type	Description
Iteration Identifier	String	The IterationIdentifier that contains the root sequence number that was passed in the “RootCause” token.
RootSequence Number	Uint64	The object root cause id.

Return values

The PositionToFirstRecordRoot() method returns the following error code:

- 0: The IterationIdentifier was successfully created.
- 5: One of the mandatory parameters is missing.

PositionToFirstRecordType()

The PositionToFirstRecordType() method creates an IterationIdentifier whose SequenceNumber token is not set and whose other parameters are set according to the parameters that were passed in.

Parameters

The IterationIdentifier is used in subsequent GetRecord or PositionAtRecord calls.

The PositionToFirstRecordType() method belongs to the IBMTSSVC_MessageLog class. Table 227 shows the parameters you can specify for the PositionToFirstRecordType() method.

Table 227. PositionToFirstRecordType() parameters

Parameter	Type	Description
ConfigOnly	Boolean	Is required to be false or not specified.
IterationIdentifier		The IterationIdentifier that was generated.

Table 227. PositionToFirstRecordType() parameters (continued)

Parameter	Type	Description
ObjectID	Uint64	The SAN Volume Controller object id. This parameter will be set to the LSObjID token of the IterationIdentifier.
ObjectType	String	The SAN Volume Controller object type. This parameter will be set to the LSObjType token of the IterationIdentifier.
UnfixedOnly	Boolean	Display only unfixed errors.

Return values

The PositionToFirstRecordType() method returns the following error code:

- 0: The IterationIdentifier was successfully created.

Reload2062Node()

The Reload2062Node() method enables a 2062 node that has been shut down.

Parameters

The Reload2062Node() method belongs to the IBMTSSVC_Provider class.
Table 228shows the parameters you can specify for the Reload2062Node() method.

Table 228. Reload2062Node() parameters

Parameter	Type	Description
Node	Uint8	The node on the 2062 blade that will be reset
Password	String	The password required to log on to the switch
Slot	Uint8	The slot of the 2062 blade where a node will be reset
SwitchIP	String	The IP of the switch where the 2062 blade resides
User	String	The user name required to log on to the switch

Return values

The Reload2062Node() method returns one of the following error codes:

- 0: The method completed successfully.
- 2: The method failed.
- 5. The number or type of parameters that have been passed is incorrect.
- ..: DMTF reserved.
- 0x8000: Connection to cluster refused.
- 0x8001: Syntax error in cluster name.
- 0x8002: Invalid node.
- 0x8003: Invalid user name or password.
- 0x8004: Syntax error in switch IP.

- 0x8005: Syntax error in cluster IP.
- 0x8006: Invalid slot.

RemoveAccess()

The RemoveAccess() method is used to delete a temporary IBMTSSVC_AccessControlInformation instance and its associations.

Parameters

The RemoveAccess() method belongs to the IBMTSSVC_AuthorizationService class. Table 229 shows the parameters you can specify for the RemoveAccess() method.

Table 229. RemoveAccess() parameters

Parameter	Type	Description
Subject	CIM_ManagedElement REF	Reference to a ManagedElement instance (associated through AuthorizedSubject) for which privileges are to be revoked.
Privilege	CIM_AuthorizedPrivilege REF	A reference to the AuthorizedPrivilege that is to be revoked.
Target	CIM_ManagedElement REF	A reference to a ManagedElement (associated through AuthorizedTarget) which will no longer be protected through AuthorizedPrivilege.

Return values

The RemoveAccess() method returns one of the following error codes:

- 0: All instances were successfully deleted.
- 2: An unexpected error occurred.
- 3: Timeout.
- 4: Failed.
- 5: The number or type of parameters that have been passed is incorrect.
- 6..15999: DMTF Reserved.
- 16000: Unsupported Privilege.
- 16001: Unsupported Target.
- 16002: Authorization Error.
- 16003: Null parameter not supported.
- 16004..32767: Method Reserved.
- 32768..65535: Vendor Specific.

RemoveCluster()

The RemoveCluster() method can be used to remove a SAN Volume Controller cluster from the ICAT configuration.

Parameters

The RemoveCluster() method belongs to the IBMTSSVC_Provider class. Table 230 on page 327shows the parameters you can specify for the RemoveCluster()

method.

Table 230. RemoveCluster() parameters

Parameter	Type	Description
ClusterIP	String	The IP of the cluster to be removed

Return values

The RemoveCluster() method returns one of the following error codes:

- 0: The method completed successfully.
- 5. The number or type of parameters that have been passed is incorrect.
- ..: DMTF reserved.

RequestDiscovery()

The RequestDiscovery() method is used to initiate a re-scan of the fibre-channel SAN to discover any new LUNs.

Parameters

The RequestDiscovery() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 231 shows the parameters you can specify for the RequestDiscovery() method.

Table 231. RequestDiscovery() parameters

Parameter	Type	Description
DiscoveredElement Count	Uint32	The number of discovered LUNs (BackendVolumes).
DiscoveredElement Instances	String[]	String representations of the instances of the discovered LUNs (BackendVolumes).
DiscoveredElements	String[]	String representations of the COPs of the discovered LUNs (BackendVolumes).

Return values

The RequestDiscovery() method returns one of the following error codes:

- 0: Discovery successfully invoked.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

Reset2062Node()

The Reset2062Node() method removes the cached data about a cluster from a 2062 node.

Parameters

The Reset2062Node() method belongs to the IBMTSSVC_Provider class. Table 232 on page 328 shows the parameters you can specify for the Reset2062Node() method.

Table 232. Reset2062Node() parameters

Parameter	Type	Description
Node	Uint8	The node on the 2062 blade that will be reset
Password	String	The password required to log on to the switch
Slot	Uint8	The slot of the 2062 blade where a node will be reset
SwitchIP	String	The IP of the switch where the 2062 blade resides
User	String	The user name required to log on to the switch

Return values

The Reset2062Node() method returns one of the following error codes:

- 0: The method completed successfully.
- 2: The method failed.
- 5. The number or type of parameters that have been passed is incorrect.
- ..: DMTF reserved.
- 0x8000: Connection to cluster refused.
- 0x8001: Syntax error in cluster name.
- 0x8002: Invalid node.
- 0x8003: Invalid user name or password.
- 0x8004: Syntax error in switch IP.
- 0x8005: Syntax error in cluster IP.
- 0x8006: Invalid slot.

RestoreConfiguration()

The RestoreConfiguration() command executes the configuration restore script.

Parameters

The script restores the current cluster configuration associated with the current instance of ClusteringService, from the corresponding cluster configuration backup.

Table 233 shows the parameters you can specify for the RestoreConfiguration() method.

Table 233. RestoreConfiguration() parameters

Parameter	Type	Description
Force	Boolean	True/False statement. If true, command continuation on non-severe errors/warnings is forced. False is the default.
Format	Boolean	True/False statement. If true, the vdisk is formatted during restore. False is the default.
Phase	Uint8	The phase to be performed. Any value other than 1 (prepare) or 2 (execute) will result in rc 5.

Table 233. RestoreConfiguration() parameters (continued)

Parameter	Type	Description
Messages	String[]	The errors/warnings received from the backup script.

Return values

The RestoreConfiguration() method returns one of the following error codes:

- 0: The backup was successful.
- 2: An unexpected error occurred and the command failed.
- 5: The number or type of passed parameters is incorrect.
- 0x8000: The specified backup was not found.
- 0x8001: The backup script returned with an error.
- 0x8002: The backup file upload via scp failed.
- 0x8003: The cluster's /tmp/dir is cleared of any backups before uploading the backup file. The command has failed.

ReturnToStoragePool()

The ReturnToStoragePool() method is used to delete an IBMTSSVC_StorageVolume instance if it is not mapped to any host.

Parameters

The ReturnToStoragePool() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 234 shows the parameters you can use for the ReturnToStoragePool() method.

Table 234. ReturnToStoragePool() parameters

Parameter	Type	Description
[Force]	Boolean	If set to True the, the volume will be deleted, ignoring the usual constraints (for example, if no LUN mapping exists for the volume). The default is False.
Job	CIM_ConcreteJob REF	Set to null.
TheElement	CIM_LogicalElement REF	The element to be returned to the pool. Must be a StorageVolume that belongs to the same cluster as the StorageConfigurationService.

Return values

The ReturnToStoragePool() method returns one of the following error codes:

- 0: The volume was successfully deleted.
- 2: An unknown error occurred.
- 3: The action timed out.
- 4: An unexpected error occurred.
- 5: At least one of the parameters was invalid.
- 6: The method is in use.

- 4096: Method parameters checked. Job started.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

SetIOGroup()

The SetIOGroup() method assigns a StorageVolume to another I/O group.

Parameters

The SetIOGroup() method belongs to the IBMTSSVC_ControllerConfigurationService class. Table 235 shows the parameters you can specify for the SetIOGroup() method:

Table 235. SetIOGroup() parameters

Parameter	Type	Description
Force	Boolean	Set to True in order to move a Volume to or from the recovery I/O group.
Group	IBMTSSVC_IOGroup REF	The IOGroup to which to assign the StorageVolume.
Volume	IBMTSSVC_StorageVolume REF	The StorageVolume to move.

Return values

The SetIOGroup() method returns one of the following error codes:

- 0: The volume was successfully moved.
- 2: The action failed.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: The I/O group must have the nodes aggregated.
- 0x8100: One or more parameters were out of cluster scope.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes.

SetLocale()

The SetLocale() method is used to define the locale of the specified cluster.

Parameters

The SetLocale() method belongs to the IBMTSSVC_Cluster class. Table 236 shows the parameters you can specify for the SetLocale() method.

Table 236. SetLocale() parameters

Parameter	Type	Description
Locale	Uint16	Sets the locale value of the named cluster; valid values are US English, Simplified Chinese, Traditional Chinese, Japanese, Korean, French, German, Italian, Spanish, and Portuguese

Return values

The SetLocale() method returns one of the following error codes:

- 0: The locale was set.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000 : The submitted locale was invalid (greater than 9).
- 0x9000 to 0x9FFF: SAN Volume Controller CLI return codes.

SetPasswords()

The SetPasswords() method is used to set the passwords for the administrators and service personnel to access an IBMTSSVC_Cluster.

Parameters

The SetPasswords() method belongs to the IBMTSSVC_ClusteringService class. Table 237 shows the parameters you can specify for the SetPasswords() method.

Table 237. SetPasswords() parameters

Parameter	Type	Description
[AdminPW]	String	Changes the administer's password to the cluster
[ServicePW]	String	Changes the service personnel's password to the cluster

Return values

The SetPasswords() method returns one of the following error codes:

- 0: The passwords were changed.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

SetQuorum()

The SetQuorum() method is used to identify an IBMTSSVC_BackendVolume as a quorum volume.

Parameters

The SetQuorum() method belongs to the IBMTSSVC_StorageConfigurationService class. Table 238 shows the parameters you can specify for the SetQuorum() method.

Table 238. SetQuorum() parameters

Parameter	Type	Description
Volume	IBMTSSVC_Backend Volume REF	Defines the IBMTSSVC_BackendVolume as a quorum disk. Must belong to the same Cluster as the StorageConfigurationService.
QuorumID	Uint8	Specifies the ID of the quorum volume, 0, 1, or 2.

Return values

The SetQuorum() method returns one of the following error codes:

- 0: The quorum volume was established.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: The quorum ID is a number greater than 2.
- 0x8100: The volume belongs to a different cluster than the StorageConfigurationService.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

SetTimeZone()

The SetTimeZone() method is used to specify the time zone of an IBMTSSVC_Cluster instance.

Parameters

The SetTimeZone() method belongs to the IBMTSSVC_ClusteringService class. Table 239 shows the parameters you can specify for the SetTimeZone() method.

Table 239. SetTimeZone() parameters

Parameter	Type	Description
Zone	COP	Defines the name of the TimeZone to set through an IBMTSSVC_AvailableTimeZone association to the IBMTSSVC_ClusteringService instance

Return values

The SetTimeZone() method returns one of the following error codes:

- 0: The time zone for the cluster was successfully set.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: The submitted time zone is not associated to the clustering service.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

StartStatisticsCollection()

The StartStatisticsCollection() method is used to initiate the statistics collection about an IBMTSSVC_Cluster instance.

Parameters

The StartStatisticsCollection() method belongs to the IBMTSSVC_ClusteringService class. Table 240 shows the parameters you can specify for the StartStatisticsCollection() method.

Table 240. StartStatisticsCollection() parameters

Parameter	Type	Description
Interval	Uint32	Sets the time interval in minutes for gathering the statistics of the cluster

Return values

The StartStatisticsCollection() method returns one of the following error codes:

- 0: The collection was started.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

StopStatisticsCollection()

The StopStatisticsCollection() method is used to terminate the statistics collection about an IBMTSSVC_Cluster instance.

Return values

The StopStatisticsCollection() method belongs to the IBMTSSVC_ClusteringService class.

The StopStatisticsCollection() method returns one of the following error codes:

- 0: The collection was stopped.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

Shutdown()

The Shutdown() method is used to shut down an IBMTSSVC_Node instance or an IBMTSSVC_Cluster instance.

Parameters

The Shutdown() method belongs to the IBMTSSVC_ClusteringService class.Table 241 shows the parameters you can specify for the Shutdown() method.

Table 241. Shutdown() parameters

Parameter	Type	Description
System	COP	Specifies the IBMTSSVC_Cluster or IBMTSSVC_Node to be shut down
[Force]	Boolean	If set to True, shuts down the remaining online node of an IBMTSSVC_RedundancyGroup

Return values

The Shutdown() method returns one of the following error codes:

- 0: The shutdown for the node or cluster was successfully initiated.
- 2: An unexpected error occurred.
- 5: The number or type of parameters that have been passed is incorrect.
- 0x8000: The submitted computer system was not of type IBMTSSVC_Node or IBMTSSVC_Cluster.
- 0x8100: One of the parameters was out of the cluster scope.
- 0x9000 to 0x9FFF: SAN Volume Controller return codes and messages.

UnfixRecord()

The UnfixRecord() method unfixes one entry in the log (marks the entry as not fixed).

Parameters

The UnfixRecord() method belongs to the IBMTSSVC_MessageLog class.
Table 242 shows the parameters you can specify for the UnfixRecord() method.

Table 242. UnfixRecord() parameters

Parameter	Type	Description
RecordNumber	Uint64	Is passed to the CLI command in the -u parameter.

Return values

The UnfixRecord() method returns the following error codes:

- 0: The method successfully unfixed the entry.
- 2: A command failed.
- 5: One of the mandatory parameters is missing.
- 0x9000 to 0x9FFF: This range represents various SAN Volume Controller return codes and messages.

Upgrade()

The Upgrade() method upgrades the software of one SAN Volume Controller node in service mode.

Parameters

The Upgrade() method belongs to the IBMTSSVC_ServiceModeService class.
Table 243 shows the parameters you can specify for the Upgrade() method.

Table 243. Upgrade() parameters

Parameter	Type	Description
FilePath	String	The location where the new software is stored.
SMNode	IBMTSSVC_Service ModeNode REF	Specifies the node that is in service mode that will be upgraded.

Return values

The Upgrade() method returns one of the following error codes:

- 0: The Upgrade() method was successful.
- 2: The Upgrade() method failed.
- 5: The wrong number or type of parameter has been passed.
- ..: DMTF reserved.
- 0x9000 to 0x9FFF: Device error codes.

WriteRecord()

The WriteRecord() method is an unsupported method that belongs to the IBMTSSVC_MessageLog class.

Parameters

Table 244 shows the parameters you can specify for the WriteRecord() method.

Table 244. WriteRecord() parameters

Parameter	Type	Description
IterationIdentifier	String	
PositionToNext	Boolean	Boolean value indicating that the IterationIdentifier should be advanced to the next record after writing the Log entry.
RecordData	Uint8[]	
RecordNumber	Uint64	

Return values

The WriteRecord() method returns one of the following error codes:

- 0: The method completed successfully.
- 2: Unknown.
- 3: Timeout.
- 4: Failed.
- 5. The number or type of parameters that have been passed is incorrect.

Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully.

Features

These are the major accessibility features in the SAN Volume Controller master console:

- You can use screen-reader software and a digital speech synthesizer to hear what is displayed on the screen. The following screen readers have been tested: JAWS v4.5 and IBM Home Page Reader v3.0.
- You can operate all features using the keyboard instead of the mouse.

Navigating by keyboard

You can use keys or key combinations to perform operations and initiate many menu actions that can also be done through mouse actions. You can navigate the SAN Volume Controller Console and help system from the keyboard by using the following key combinations:

- To traverse to the next link, button, or topic, press Tab inside a frame (page).
- To expand or collapse a tree node, press →or ←, respectively.
- To move to the next topic node, press V or Tab.
- To move to the previous topic node, press ^ or Shift+Tab.
- To scroll all the way up or down, press Home or End, respectively.
- To go back, press Alt+←.
- To go forward, press Alt+→.
- To go to the next frame, press Ctrl+Tab.
- To move to the previous frame, press Shift+Ctrl+Tab.
- To print the current page or active frame, press Ctrl+P.
- To select, press Enter.

Accessing the publications

You can view the publications for the SAN Volume Controller in Adobe Portable Document Format (PDF) using the Adobe Acrobat Reader. The PDFs are provided on a CD that is packaged with the product or you can access them at the following Web site:

http://www-1.ibm.com/servers/storage/support/virtual/2145.html

Related reference

“SAN Volume Controller library and related publications” on page xviii
A list of other publications that are related to this product are provided to you for your reference.

© Copyright IBM Corp. 2003, 2004

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATIONS ″AS IS″ WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

© Copyright IBM Corp. 2003, 2004

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

- FlashCopy
- IBM
- TotalStorage

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Glossary

Glossary of terms used in the CIM Agent Developer's Reference Guide.

A

association
A class that contains two references that define a relationship between two referenced objects.

agent code
An open-systems standard that interprets Common Information Model (CIM) requests and responses as they transfer between the client application and the device.

C

CIM See *Common Information Model*.

CIMOM
See *CIM object manager*.

Common Information Model (CIM)
A set of standards developed by the Distributed Management Task Force (DMTF). CIM provides a conceptual framework for storage management and an open approach to the design and implementation of storage systems, applications, databases, networks, and devices.

class The definition of an object within a specific hierarchy. A class can have properties and methods and can serve as the target of an association.

client application
A storage management program that initiates Common Information Model (CIM) requests to the CIM agent for the device.

CIMOM
See *CIM object manager*.

CIM object manager (CIMOM)
The common conceptual framework for data management that receives, validates, and authenticates the CIM requests from the client application. It then directs the requests to the appropriate component or service provider.

D

device provider
A device-specific handler that serves as a plug-in for the Common Information Model (CIM); that is, the CIM object manager (CIMOM) uses the handler to interface with the device.

device

- In the CIM Agent, the storage server that processes and hosts client application requests.
- IBM definition: A piece of equipment that is used with the computer and does not generally interact directly with the system, but is controlled by a controller.
- HP definition: In its physical form, a magnetic disk that can be attached to a SCSI bus. The term is also used to indicate a physical device that has been made part of a controller configuration; that is, a physical

© Copyright IBM Corp. 2003, 2004

device that is known to the controller. Units (virtual disks) can be created from devices after the devices have been made known to the controller.

I

indication
An object representation of an event.

instance
An individual object that is a member of some class. In object-oriented programming, an object is created by instantiating a class.

M

method
A way to implement a function on a class.

N

namespace
The scope within which a Common Information Model (CIM) schema applies.

O

object In object-oriented design or programming, a concrete realization of a class that consists of data and the operations associated with that data.

object model
A representation, such as a diagram, of objects in a given system. Using symbols similar to standard flowchart symbols, an object model depicts the classes the objects belong to, their associations with each other, the attributes that make them unique, and the operations that the objects can perform and that can be performed on them.

object name
An object that consists of a namespace path and a model path. The namespace path provides access to the Common Information Model (CIM) implementation managed by the CIM Agent, and the model path provides navigation within the implementation.

P

property
In the Common Information Model (CIM), an attribute that is used to characterize instances of a class.

Q

qualifier
A value that provides additional information about a class, association, indication, method, method parameter, instance, property, or reference.

R

reference
A pointer to another instance that defines the role and scope of an object in an association.

S

SMI-S See *Storage Management Initiative Specification*.

Storage Management Initiative Specification (SMI-S)
A design specification developed by the Storage Networking Industry Association (SNIA) that specifies a secure and reliable interface that allows storage management systems to identify, classify, monitor, and control physical and logical resources in a storage area network. The interface is intended as a solution that integrates the various devices to be managed in a storage area network (SAN) and the tools used to manage them.

Service Location Protocol (SLP)
In the Internet suite of protocols, a protocol that identifies and uses network hosts without having to designate a specific network host name.

schema
A group of object classes defined for and applicable to a single namespace. Within the CIM Agent, the supported schemas are the ones that are loaded through the managed object format (MOF).

W

WBEM
See *Web-Based Enterprise Management*.

Web-Based Enterprise Management (WBEM)
A tiered, enterprise-management architecture that was developed by the Distributed Management Task Force (DMTF). This architecture provides the management design framework that consists of devices, device providers, the object manager, and the messaging protocol for the communication between client applications and the object manager.

Index

A

accessibility
  keyboard 337
  shortcut keys 337
adding
  candidate node 27
  nodes 27
association object classes 228
  AccountManagementServiceForSystem 264
  AccountOnCIMOM 265
  AccountOnSystem 265
  AllocatedFromStoragePool 228
  AuthorizationTarget 230
  AuthorizedCollection 229
  AuthorizedStorageHardwareID 229
  AuthorizedSubject 230
  AvailableHardwareID 230
  BackendControllerForVolume 231
  BasedOn 231
  ClusterController 232
  ClusterDumps 232
  ClusteringCandidate 233
  ClusteringServiceForSystem 233
  ClusterMaskingCapabilities 234
  ClusterPort 234
  ClusterScopeCandidateVolume 234
  ClusterScopeChassis 235
  ClusterScopeFCSet 235
  ClusterScopeIOGroup 235
  ClusterScopeNodeVPD 236
  ClusterScopePrivilege 236
  ClusterScopeProduct 236
  ClusterScopeSCSet 236
  ComponentCS 237
  ComputerSystemPackage 237
  ConnectedBackendController 238
  ControllerConfigurationServiceForSystem 238
  ControllerConfServiceMaskingCapabilities 239
  CopyCandidate 239
  ElementConformsToProfile 239
  FlashCopyStorageSynchronized 240
  FlashCopySynchronizedMember 242
  HardwareIDOnSystem 243
  HostedAccessPoint 243
  HostedJob 243
  HostedPrimordialPool 243
  HostedStoragePool 244
  HwIDCollectionOnSystem 244
  IBMTS_CommMechanismForManager 266
  IBMTS_ElementConformsToProfile 266
  IBMTS_NamespaceInManager 268
  IBMTSSVC_ClusterVolume 237
  IBMTSSVC_SystemFCPort 261
  IndicationFiltersConformsToProfile 244
  IndicationFiltersConformsToSubProfile 245
  IOGroupIdentity 245
  IOGroupPort 245
association object classes *(continued)*
  ManagesAccount 268
  ManagesCollection 246
  ManagesController 246
  ManagesHardwareId 246
  ManagesPrivilege 247
  MemberOfCollection 247
  MemberOfIOGroup 247
  NodeDumps 248
  PartnershipCandidate 249
  PoolCapabilities 249
  PrimordialPoolCapabilities 249
  PrimordialPoolComponent 250
  PrimordialPoolForController 250
  PrivilegeServiceForSystem 250
  ProductPhysicalComponent 251
  ProtocolControllerForPort 251
  ProtocolControllerForUnit 252
  ProviderInObjectManager 253
  RemotePartnership 254
  RemoteSystemVolume 254
  RequiresProfile 254
  SAPAvailableForElement 255
  StorageConfigurationServiceCap 255
  StorageConfigurationServiceForSystem 255
  StorageHardwareIDManagement ServiceForSystem 256
  StoragePoolComponent 256
  SyncCopyStorageSynchronized 256
  SyncCopySynchronizedMember 260
  SystemBackendVolume 260
  SystemCandidateVolume 260
  SystemController 261
  SystemFeatures 261
  SystemVolume 261
  SystemVPD 262
  UseOfMessageLog 263
  VolumeSettingData 263

C

candidate node
  adding 27
certificate
  truststore 9
CIM (Common Information Model) 2
  agent 3
CIM agent 8
  CIMOM 3
  device 3
  device provider 3
  functional views
    clustering service 16
    copy services 20
    LUN masking 18
    security service 23
  functional views of 10
    access point subprofile 15

© Copyright IBM Corp. 2003, 2004

CIM agent *(continued)*
functional views of *(continued)*
cluster subprofile 16
copy services 20
extent mapping subprofile 13
ExtraCapacitySet subprofile 14
Flash Copy state diagram 34
LUN creation subprofile 19
LUN masking 18
physical package 11
pool manipulation 24
profile overview 10
security services 23
server profile 12
Sync Copy state diagram 35
vendor specific service mode subprofile 21
vendor-specific cluster operations 22
vendor-specific storage configuration operations 17
IP discovery 39
IP registration 39
methods 269
object classes 41
association 228
core 41
security 206
SLP based discovery 39
CIMOM
manual SLP registration 39
Cisco Systems
MDS 9000 Caching Services Module 5
MDS 9000 switch 5
clustering
service 16
Common Information Model (CIM) 2
agent 3
configuration
adding a candidate node to a cluster 27
creating a new storage pool 28
creating a new storage volume 29
modifying a storage pool 28
performing basic storage configuration 27
storage 27
storage configuration 27
configuring
storage 27
conventions
emphasis in text xviii
copy services 31
overview 20
core object classes
BackendController 41
BackendVolume 49
CandidateCluster 59
CandidateNode 60
CandidateStorageHardwareID 67
CandidateVolume 68
Chassis 69
Cluster 77
Controller 84
ControllerConfigurationService 91
core object classes *(continued)*
ControllerMaskingCapabilities 95
Dumps 97
FCPort 98
Features 108
FlashCopySynchronizedSet 109
HardwareIdCollection 111
IOGroup 112
IOGroupSet 118
Job 119
MessageLog 124
Node 132
NodeVPD 138
overview 41
PrimordialStoragePool 138
Privilege 140
PrivilegeManagementService 195
Product 142
Provider 143
RegisteredProfile 146
RegisteredSubProfile 149
RemoteCluster 152
RemoteServiceAccessPoint 154
RemoteVolume 160
StorageCapabilities 161
StorageConfigurationCapabilities 164
StorageConfigurationService 199
StorageHardwareIDManagementService 202
StoragePool 169
StorageSetting 172
StorageVolume 175
SyncCopySynchonronizedSet 189
creating
FlashCopy
relationship 31
storage
pools 28
volumes 29
synchronous copies
relationship 31
synchronous copy relationships
between volumes in different clusters 33
between volumes in the same cluster 33

E

emphasis in text xviii
expiration 9
expired
certificate 9
extrinsic methods
Add2062Cluster() 282
Add2145Cluster() 283
AddHardwareIDsToCollection() 283
AddNode() 284
AssignAccess() 285
AttachDevice() 287
AttachReplica 287
BackupConfiguration() 288
CancelIteration() 289
Clean() 289

extrinsic methods *(continued)*
ClearLog() 290
Create2062Cluster() 290
CreateHardwareIDCollection() 291
CreateOrModifyElementFromStoragePool() 295
CreateOrModifyStoragePool() 292
CreateProtocolControllerWithPorts() 297
CreateRemoteClusterPartnership() 298
CreateReplica() 299
CreateSetting() 300
CreateStorageHardwareID() 300
CreateSynchronizedSet() 301
DeleteConfigurationBackup() 302
DeleteHardwareIDCollection() 303
DeleteProtocolController() 303
DeleteRecord() 304
DeleteRemoteClusterPartnership() 305
DeleteStorageHardwareID() 305
DeleteStoragePool() 307
DeleteSynchronizedSet() 306
DetachDevice() 308
Dump() 308
Enter() 309
EvictNode() 309
Exit() 310, 334
FixRecord() 310
GetAllRecords() 311
GetDump() 311
GetFreeExtents() 312
GetRecord() 313
GetResetPasswordChangeFeatureStatus() 313
GetSupportedSizeRange() 314
IncludeBackendVolume() 315
ListConfigurationBackups() 316
MigrateVolume() 316
ModifyErrorSettings() 317
ModifyIPAddress() 318
ModifyResetPasswordChangeFeature() 318
ModifySynchronization() 318
ModifySynchronizedSet() 320
overview 279
PositionToFirstRecord() 323
PositionToFirstRecordRoot() 324
PositionToFirstRecordType() 323, 324
Reload2062Node() 325
RemoveAccess() 326
RemoveCluster() 326
RequestDiscovery() 327
Reset2062Node() 327
RestoreConfiguration() 328
ReturnToStoragePool() 329
SetIOGroup() 330
SetLocale() 330
SetPasswords() 331
SetQuorum() 331
SetTimeZone() 332
Shutdown() 333
StartStatisticsCollection() 332
StopStatisticsCollection() 333
UnfixRecord() 334
WriteRecord() 335

F

FlashCopy
creating
synchronized set 32
FlashCopy service 31
functional views of the CIM Agent 10
access point subprofile 15
cluster subprofile 16
copy services 20
extent mapping subprofile 13
ExtraCapacitySet subprofile 14
Flash Copy state diagram 34
LUN creation subprofile 19
LUN masking 18
physical package 11
pool manipulation 24
profile overview 10
security service 23
server profile 12
Sync Copy state diagram 35
vendor specific service mode subprofile 21
vendor-specific cluster operations 22
vendor-specific storage configuration operations 17

G

generating new truststore certificate 9

I

IBMTS object classes
Account 206
AccountManagementService 208
AccountOnSystem 265
CIMXMLCommunicationMechanism 212
IBMTS_CommMechanismForManager 266
IBMTS_ElementConformsToProfile 266
IBMTS_IndicationFilter 218
IBMTS_NameSpace 218
IBMTS_NamespaceInManager 268
ManagesAccount 268
ObjectManager 220
RegisteredProfile 224
System 225
IBMTSSVC object classes
AccountManagementServiceForSystem 264
AccountOnCIMOM 265
AllocatedFromStoragePool 228
AuthorizedCollection 229
AuthorizedStorageHardwareID 229
AuthorizedSubject 230
AuthorizedTarget 230
AvailableHardwareID 230
BackendController 41
BackendControllerForVolume 231
BackendVolume 49
BasedOn 231
CandidateCluster 59
CandidateNode 60
CandidateStorageHardwareID 67

IBMTSSVC object classes *(continued)*
CandidateVolume 68
Chassis 69
Cluster 77
ClusterController 232
ClusterDumps 232
ClusteringCandidate 233
ClusteringService 191
ClusteringServiceForSystem 233
ClusterMaskingCapabilities 234
ClusterPort 234
ClusterScopeCandidateVolume 234
ClusterScopeChassis 235
ClusterScopeFCSet 235
ClusterScopeIOGroup 235
ClusterScopeNodeVPD 236
ClusterScopePrivilege 236
ClusterScopeProduct 236
ClusterScopeSCSet 236
ComponentCS 237
ComputerSystemPackage 237
ConnectedBackendController 238
Controller 84
ControllerConfigurationService 91
ControllerConfigurationServiceForSystem 238
ControllerConfServiceMaskingCapabilities 239
ControllerMaskingCapabilities 95
CopyCandidate 239
Dumps 97
ElementConformsToProfile 239
FCPort 98
Features 108
FlashCopyStorageSynchronized 240
FlashCopySynchronizedMember 242
FlashCopySynchronizedSet 109
HardwareIdCollection 111
HardwareIDOnSystem 243
HostedAccessPoint 243
HostedJob 243
HostedPrimordialPool 243
HostedStoragePool 244
HwIDCollectionOnSystem 244
IBMTSSVC_ClusterVolume 237
IBMTSSVC_SystemFCPort 261
IndicationFiltersConformsToProfile 244
IndicationFiltersConformsToSubProfile 245
IOGroup 112
IOGroupIdentity 245
IOGroupPort 245
IOGroupSet 118
Job 119
ManagesCollection 246
ManagesController 246
ManagesHardwareId 246
ManagesPrivilege 247
MemberOfCollection 247
MemberOfIOGroup 247
MessageLog 124
Node 132
NodeDumps 248
NodeVPD 138

IBMTSSVC object classes *(continued)*
PartnershipCandidate 249
PoolCapabilities 249
PrimordialPoolCapabilities 249
PrimordialPoolComponent 250
PrimordialPoolForController 250
PrimordialStoragePool 138
Privilege 140
PrivilegeManagementService 195
PrivilegeServiceForSystem 250
Product 142
ProductPhysicalComponent 251
ProtocolControllerForPort 251
ProtocolControllerForUnit 252
Provider 143
ProviderInObjectManager 253
RegisteredProfile 146
RegisteredSubProfile 149
RemoteCluster 152
RemotePartnership 254
RemoteServiceAccessPoint 154
RemoteSystemVolume 254
RemoteVolume 160
RequiresProfile 254
SAPAvailableForElement 255
StorageCapabilities 161
StorageConfigurationCapabilities 164
StorageConfigurationService 199
StorageConfigurationServiceCap 255
StorageConfigurationServiceForSystem 255
StorageHardwareIDManagement ServiceForSystem 256
StorageHardwareIDManagementService 202
StoragePool 169
StoragePoolComponent 256
StorageSetting 172
StorageVolume 175
SyncCopyStorageSynchronized 256
SyncCopySynchronizedMember 260
SyncCopySynchronizedSet 189
SystemBackendVolume 260
SystemCandidateVolume 260
SystemController 261
SystemFeatures 261
SystemVolume 261
SystemVPD 262
UseOfMessageLog 263
VolumeSettingData 263
information
center xviii
intrinsic methods
Associators() 269
AssociatorsNames() 270
CreateInstance() 271
DeleteInstance() 272
EnumerateClasses() 272
EnumerateClassNames() 273
EnumerateInstanceNames() 274
EnumerateInstances() 274
ExecQuery() 275
GetClass() 275

intrinsic methods *(continued)*
GetInstance() 276
GetProperty() 276
ModifyInstance() 277
overview 269
Reference() 278
ReferenceNames() 278
SetProperty() 279

K

keyboard 337
shortcut keys 337

L

logical unit numbers (LUNs)
masking 18
LUN masking
performing 37
LUNs (logical unit numbers)
masking 18, 37

M

Management Application
launching Web User Interface 39
masking LUNs 37
master console
error 9
methods
CIM Agent 269
extrinsic 279
Add2062Cluster() 282
Add2145Cluster() 283
AddHardwareIDsToCollection() 283
AddNode() 284
AssignAccess() 285
AttachDevice() 287
AttachReplica 287
BackupConfiguration() 288
CancelIteration() 289
Clean() 289
ClearLog() 290
Create2062Cluster() 290
CreateHardwareIDCollection() 291
CreateOrModifyElementFromStoragePool() 295
CreateOrModifyStoragePool() 292
CreateProtocolControllerWithPorts() 297
CreateRemoteClusterPartnership() 298
CreateReplica() 299
CreateSetting() 300
CreateStorageHardwareID() 300
CreateSynchronizedSet() 301
DeleteConfigurationBackup() 302
DeleteHardwareIDCollection() 303
DeleteProtocolController() 303
DeleteRecord() 304
DeleteRemoteClusterPartnership() 305
DeleteStorageHardwareID() 305
DeleteStoragePool() 307
methods *(continued)*
extrinsic *(continued)*
DeleteSynchronizedSet() 306
DetachDevice() 308
Dump() 308
Enter() 309
EvictNode() 309
Exit() 310, 334
FixRecord() 310
GetAllRecords() 311
GetDump() 311
GetFreeExtents() 312
GetRecord() 313
GetResetPasswordChangeFeatureStatus() 313
GetSupportedSizeRange() 314
IncludeBackendVolume() 315
ListConfigurationBackups() 316
MigrateVolume() 316
ModifyErrorSettings() 317
ModifyIPAddress() 318
ModifyResetPasswordChangeFeature() 318
ModifySynchronization() 318
ModifySynchronizedSet() 320
PositionToFirstRecord() 323
PositionToFirstRecordRoot() 324
PositionToFirstRecordType() 323, 324
Reload2062Node() 325
RemoveAccess() 326
RemoveCluster() 326
RequestDiscovery() 327
Reset2062Node() 327
RestoreConfiguration() 328
ReturnToStoragePool() 329
SetIOGroup() 330
SetLocale() 330
SetPasswords() 331
SetQuorum() 331
SetTimeZone() 332
Shutdown() 333
StartStatisticsCollection() 332
StopStatisticsCollection() 333
UnfixRecord() 334
WriteRecord() 335
intrinsic 269
AssociatorNames() 270
Associators() 269
CreateInstance() 271
DeleteInstance() 272
EnumerateClasses() 272
EnumerateClassNames() 273
EnumerateInstanceNames() 274
EnumerateInstances() 274
ExecQuery() 275
GetClass() 275
GetInstance() 276
GetProperty() 276
ModifyInstance() 277
Reference() 278
ReferenceNames() 278
SetProperty() 279
mkcertificate.bat 9

modifying
storage pools 28
Multiple network cards 39

N

nodes
adding 27
notices
legal 339

O

object classes
association 228
core 41
overview 41
security 206
service 191
ordering publications xx

P

performing
LUN masking 37
publications
ordering xx

R

related information xviii
RemoteServiceAccessPoint
manually set connection data 39

S

SAN Volume Controller
overview 5
security
object classes 206
Account 206
AccountManagementService 208
CIMXMLCommunicationMechanism 212
IBMTS_IndicationFilter 218
IBMTS_NameSpace 218
ObjectManager 220
RegisteredProfile 224
System 225
service 23
service
object classes 191
ClusteringService 191
StorageHardwareID 168
shortcut keys 337
SMI-S (Storage Management Initiative Specification) 1
Specification, Storage Management Initiative 1
storage
configuration 27
adding a candidate node to a cluster 27
creating a new storage pool 28
storage *(continued)*
configuration *(continued)*
creating a new storage volume 29
modifying a storage pool 28
performing basic storage configuration 27
storage configuration 27
adding a candidate node to a cluster 27
creating a new storage pool 28
creating a new storage volume 29
modifying a storage pool 28
performing basic storage configuration 27
Storage Management Initiative Specification (SMI-S) 1
storage pools
creating 28
modifying 28
support
Web sites xx
synchronous copy
creating relationships
between volumes in different clusters 33
between volumes in the same cluster 33
synchronous copy service 31

T

text emphasis xviii
trademarks 340
truststore
certificate 9
truststore certificate 9

V

volumes
creating new storage 29

W

Web sites xx

Readers' Comments — We'd Like to Hear from You

IBM TotalStorage SAN Volume Controller
CIM Agent Developer's Reference
Version 1.2.1

Publication No. SC26-7545-02

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC26-7545-02

IBM®

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Information Development
Department 61C
9032 South Rita Road
Tucson, Arizona
USA 85775-4401

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

Printed in USA

SC26-7545-02

Spine information:

IBM TotalStorage SAN Volume Controller

SAN Volume Controller CIM Agent Developer's Reference

Version 1.2.1