

[MS-MQDSSM]: Message Queuing (MSMQ): Directory Service Schema Mapping

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [Open Specification Promise](#) or the [Community Promise](#). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

| Date | Revision History | Revision Class | Comments |
|------------|------------------|----------------|---|
| 07/25/2008 | 0.1 | Editorial | Initial Availability. |
| 08/29/2008 | 1.0 | Major | Added section 2.3. |
| 10/24/2008 | 2.0 | Major | Updated and revised the technical content. |
| 12/05/2008 | 2.0.1 | Editorial | Revised and edited the technical content. |
| 01/16/2009 | 2.0.2 | Editorial | Revised and edited the technical content. |
| 02/27/2009 | 2.1 | Minor | Updated the technical content. |
| 04/10/2009 | 2.2 | Minor | Updated the technical content. |
| 05/22/2009 | 2.3 | Minor | Updated the technical content. |
| 07/02/2009 | 2.3.1 | Editorial | Revised and edited the technical content. |
| 08/14/2009 | 2.3.2 | Editorial | Revised and edited the technical content. |
| 09/25/2009 | 3.0 | Major | Updated and revised the technical content. |
| 11/06/2009 | 3.0.1 | Editorial | Revised and edited the technical content. |
| 12/18/2009 | 4.0 | Major | Updated and revised the technical content. |
| 01/29/2010 | 5.0 | Major | Updated and revised the technical content. |
| 03/12/2010 | 6.0 | Major | Updated and revised the technical content. |
| 04/23/2010 | 6.0.1 | Editorial | Revised and edited the technical content. |
| 06/04/2010 | 7.0 | Major | Updated and revised the technical content. |
| 07/16/2010 | 8.0 | Major | Significantly changed the technical content. |
| 08/27/2010 | 9.0 | Major | Significantly changed the technical content. |
| 10/08/2010 | 10.0 | Major | Significantly changed the technical content. |
| 11/19/2010 | 11.0 | Major | Significantly changed the technical content. |
| 01/07/2011 | 12.0 | Major | Significantly changed the technical content. |
| 02/11/2011 | 13.0 | Major | Significantly changed the technical content. |
| 03/25/2011 | 14.0 | Major | Significantly changed the technical content. |
| 05/06/2011 | 15.0 | Major | Significantly changed the technical content. |
| 06/17/2011 | 15.1 | Minor | Clarified the meaning of the technical content. |

| Date | Revision History | Revision Class | Comments |
|-------------|-------------------------|-----------------------|--|
| 09/23/2011 | 15.1 | No change | No changes to the meaning, language, or formatting of the technical content. |
| 12/16/2011 | 16.0 | Major | Significantly changed the technical content. |
| 03/30/2012 | 16.1 | Minor | Clarified the meaning of the technical content. |
| 07/12/2012 | 16.2 | Minor | Clarified the meaning of the technical content. |
| 10/25/2012 | 17.0 | Major | Significantly changed the technical content. |
| 01/31/2013 | 17.0 | No change | No changes to the meaning, language, or formatting of the technical content. |
| 08/08/2013 | 18.0 | Major | Significantly changed the technical content. |
| 11/14/2013 | 18.0 | No change | No changes to the meaning, language, or formatting of the technical content. |

Contents

| | |
|--|-----------|
| 1 Introduction | 9 |
| 1.1 Glossary | 9 |
| 1.2 References | 9 |
| 1.2.1 Normative References | 10 |
| 1.2.2 Informative References | 10 |
| 1.3 Overview | 10 |
| 1.4 Relationship to Other Protocols | 11 |
| 1.5 Prerequisites/Preconditions | 11 |
| 1.6 Applicability Statement | 11 |
| 1.7 Versioning and Capability Negotiation | 11 |
| 1.8 Vendor-Extensible Fields | 11 |
| 1.9 Standards Assignments | 11 |
| 2 Messages | 12 |
| 2.1 Transport | 12 |
| 2.2 Common Data Types | 12 |
| 2.2.1 LDAP Distinguished Names of Directory Objects | 12 |
| 2.2.2 Attributes of Directory Objects | 13 |
| 2.2.3 mSMQEnterpriseSettings Object | 15 |
| 2.2.4 Unused Active Directory Properties | 15 |
| 2.2.5 Hash String Calculation From Queue Name | 15 |
| 2.2.6 LDAP Result Code to DirectoryOperationResult Mapping | 25 |
| 2.3 Queue Alias | 25 |
| 2.4 Directory Service Schema Elements | 25 |
| 3 Protocol Details | 28 |
| 3.1 Algorithm Details | 28 |
| 3.1.1 Abstract Data Model | 28 |
| 3.1.1.1 Externally Defined Data Elements | 28 |
| 3.1.1.2 ReadDirectoryIteratorState Data Element | 28 |
| 3.1.1.3 ReadDirectoryIteratorStateCollection Data Element | 29 |
| 3.1.1.4 User Data Element | 29 |
| 3.1.1.5 CachedConfigurationNamingContext | 29 |
| 3.1.1.6 CachedLDAPConnection | 29 |
| 3.1.2 Timers | 29 |
| 3.1.3 Initialization | 29 |
| 3.1.4 Message Processing Events and Sequencing Rules | 30 |
| 3.1.5 Timer Events | 30 |
| 3.1.6 Other Local Events | 31 |
| 3.1.6.1 Create Directory Object | 31 |
| 3.1.6.1.1 QueueManager | 31 |
| 3.1.6.1.1.1 Preconditions | 31 |
| 3.1.6.1.1.2 Creation | 32 |
| 3.1.6.1.1.3 Postprocessing | 34 |
| 3.1.6.1.1.4 mSMQOSType | 35 |
| 3.1.6.1.1.5 mSMQServiceType | 36 |
| 3.1.6.1.1.6 mSMQOutRoutingServers | 36 |
| 3.1.6.1.1.7 mSMQInRoutingServers | 37 |
| 3.1.6.1.1.8 nTSecurityDescriptor | 37 |
| 3.1.6.1.1.9 mSMQSignCertificates and mSMQDigests | 38 |

| | | |
|-------------|-------------------------------|----|
| 3.1.6.1.2 | Queue | 39 |
| 3.1.6.1.2.1 | Preconditions | 39 |
| 3.1.6.1.2.2 | Creation | 39 |
| 3.1.6.1.2.3 | Postprocessing | 40 |
| 3.1.6.1.2.4 | mSMQPrivacyLevel | 40 |
| 3.1.6.1.2.5 | Name..... | 41 |
| 3.1.6.1.3 | Site | 41 |
| 3.1.6.1.3.1 | Preconditions | 41 |
| 3.1.6.1.3.2 | Creation | 41 |
| 3.1.6.1.3.3 | Postprocessing | 42 |
| 3.1.6.1.3.4 | nTSecurityDescriptor..... | 42 |
| 3.1.6.1.4 | RoutingLink | 43 |
| 3.1.6.1.4.1 | Preconditions | 43 |
| 3.1.6.1.4.2 | Creation | 44 |
| 3.1.6.1.4.3 | Postprocessing | 45 |
| 3.1.6.1.4.4 | mSMQSite1..... | 45 |
| 3.1.6.1.4.5 | mSMQSite2..... | 45 |
| 3.1.6.1.4.6 | mSMQSiteGates | 46 |
| 3.1.6.2 | Delete Directory Object | 46 |
| 3.1.6.2.1 | QueueManager | 47 |
| 3.1.6.2.1.1 | Preconditions | 47 |
| 3.1.6.2.1.2 | Delete | 47 |
| 3.1.6.2.1.3 | Postprocessing | 47 |
| 3.1.6.2.2 | Queue | 48 |
| 3.1.6.2.2.1 | Preconditions | 48 |
| 3.1.6.2.2.2 | Delete | 48 |
| 3.1.6.2.2.3 | Postprocessing | 49 |
| 3.1.6.2.3 | Site | 49 |
| 3.1.6.2.3.1 | Preconditions | 49 |
| 3.1.6.2.3.2 | Delete | 49 |
| 3.1.6.2.3.3 | Postprocessing | 49 |
| 3.1.6.2.4 | RoutingLink | 50 |
| 3.1.6.2.4.1 | Preconditions | 50 |
| 3.1.6.2.4.2 | Delete | 50 |
| 3.1.6.2.4.3 | Postprocessing | 50 |
| 3.1.6.3 | Read Directory | 50 |
| 3.1.6.3.1 | Search For One Object..... | 51 |
| 3.1.6.3.2 | QueueManager | 52 |
| 3.1.6.3.2.1 | Preconditions | 52 |
| 3.1.6.3.2.2 | Read | 52 |
| 3.1.6.3.3 | Queue | 53 |
| 3.1.6.3.3.1 | Preconditions | 53 |
| 3.1.6.3.3.2 | Read | 53 |
| 3.1.6.3.4 | Enterprise | 54 |
| 3.1.6.3.4.1 | Preconditions | 54 |
| 3.1.6.3.4.2 | Read | 54 |
| 3.1.6.3.5 | Site | 55 |
| 3.1.6.3.5.1 | Preconditions | 55 |
| 3.1.6.3.5.2 | Read | 55 |
| 3.1.6.3.6 | RoutingLink | 56 |
| 3.1.6.3.6.1 | Preconditions | 56 |
| 3.1.6.3.6.2 | Read | 56 |
| 3.1.6.3.7 | User | 57 |

| | | |
|--------------|--|----|
| 3.1.6.3.7.1 | Preconditions | 57 |
| 3.1.6.3.7.2 | Read | 57 |
| 3.1.6.4 | Read Directory Begin | 57 |
| 3.1.6.4.1 | QueueManager | 59 |
| 3.1.6.4.1.1 | Preconditions | 59 |
| 3.1.6.4.1.2 | Read Begin | 59 |
| 3.1.6.4.1.3 | Postprocessing | 61 |
| 3.1.6.4.2 | Queue | 61 |
| 3.1.6.4.2.1 | Preconditions | 61 |
| 3.1.6.4.2.2 | Read Begin | 61 |
| 3.1.6.4.2.3 | Postprocessing | 63 |
| 3.1.6.4.3 | Enterprise | 63 |
| 3.1.6.4.3.1 | Preconditions | 63 |
| 3.1.6.4.3.2 | Read Begin | 63 |
| 3.1.6.4.3.3 | Postprocessing | 65 |
| 3.1.6.4.3.4 | WeakenedSecurity | 65 |
| 3.1.6.4.3.5 | NonLDAPCapableQueueManagerNotification | 65 |
| 3.1.6.4.4 | Site | 65 |
| 3.1.6.4.4.1 | Preconditions | 65 |
| 3.1.6.4.4.2 | Read Begin | 66 |
| 3.1.6.4.4.3 | Postprocessing | 67 |
| 3.1.6.4.5 | RoutingLink | 67 |
| 3.1.6.4.5.1 | Preconditions | 67 |
| 3.1.6.4.5.2 | Read Begin | 67 |
| 3.1.6.4.5.3 | Postprocessing | 69 |
| 3.1.6.4.5.4 | Site1Identifier Filtering | 69 |
| 3.1.6.4.5.5 | Site2Identifier Filtering | 69 |
| 3.1.6.4.6 | User | 70 |
| 3.1.6.4.6.1 | Preconditions | 70 |
| 3.1.6.4.6.2 | Read Begin | 70 |
| 3.1.6.4.6.3 | Postprocessing | 71 |
| 3.1.6.5 | Read Directory Next..... | 71 |
| 3.1.6.6 | Read Directory End | 72 |
| 3.1.6.7 | Write Directory..... | 72 |
| 3.1.6.7.1 | QueueManager | 73 |
| 3.1.6.7.1.1 | Preconditions | 73 |
| 3.1.6.7.1.2 | Write | 73 |
| 3.1.6.7.1.3 | Postprocessing | 75 |
| 3.1.6.7.1.4 | mSMQOSType | 76 |
| 3.1.6.7.1.5 | mSMQServiceType..... | 76 |
| 3.1.6.7.1.6 | mSMQOutRoutingServers | 76 |
| 3.1.6.7.1.7 | mSMQInRoutingServers | 77 |
| 3.1.6.7.1.8 | mSMQSignCertificates and mSMQDigests | 78 |
| 3.1.6.7.1.9 | mSMQSettings Objects..... | 78 |
| 3.1.6.7.1.10 | PublicSigningKeyList | 81 |
| 3.1.6.7.2 | Queue | 81 |
| 3.1.6.7.2.1 | Preconditions | 81 |
| 3.1.6.7.2.2 | Write | 81 |
| 3.1.6.7.2.3 | Postprocessing | 83 |
| 3.1.6.7.2.4 | PrivacyLevel..... | 83 |
| 3.1.6.7.2.5 | <queue name> | 83 |
| 3.1.6.7.3 | Enterprise | 84 |
| 3.1.6.7.3.1 | Preconditions | 84 |

| | | |
|--------------|--|-----|
| 3.1.6.7.3.2 | Write | 84 |
| 3.1.6.7.3.3 | Postprocessing | 85 |
| 3.1.6.7.3.4 | mSMQCSPName | 85 |
| 3.1.6.7.4 | Site | 85 |
| 3.1.6.7.4.1 | Preconditions | 85 |
| 3.1.6.7.4.2 | Write | 85 |
| 3.1.6.7.4.3 | Postprocessing | 86 |
| 3.1.6.7.5 | RoutingLink | 87 |
| 3.1.6.7.5.1 | Preconditions | 87 |
| 3.1.6.7.5.2 | Write | 87 |
| 3.1.6.7.5.3 | Postprocessing | 88 |
| 3.1.6.7.5.4 | mSMQSite1..... | 88 |
| 3.1.6.7.5.5 | mSMQSite2..... | 88 |
| 3.1.6.7.5.6 | mSMQSiteGates | 89 |
| 3.1.6.7.6 | User | 89 |
| 3.1.6.7.6.1 | Preconditions | 89 |
| 3.1.6.7.6.2 | Write | 90 |
| 3.1.6.7.6.3 | Postprocessing | 91 |
| 3.1.6.7.6.4 | Note on mSMQSignCertificates and mSMQDigests | 91 |
| 3.1.6.8 | Resolve Queue Alias..... | 91 |
| 3.1.6.9 | Resolve Distribution List | 92 |
| 3.1.6.10 | Create LDAP Attribute List..... | 93 |
| 3.1.6.10.1 | QueueManager..... | 94 |
| 3.1.6.10.2 | Queue..... | 95 |
| 3.1.6.10.3 | Enterprise | 95 |
| 3.1.6.10.4 | Site | 96 |
| 3.1.6.10.5 | RoutingLink | 96 |
| 3.1.6.10.6 | User | 97 |
| 3.1.6.11 | Create ADM Element From LDAP Values..... | 97 |
| 3.1.6.11.1 | QueueManager..... | 97 |
| 3.1.6.11.1.1 | ComputerName | 101 |
| 3.1.6.11.1.2 | OperatingSystemType | 101 |
| 3.1.6.11.1.3 | OutRoutingServerIdentifierList | 101 |
| 3.1.6.11.1.4 | InRoutingServerIdentifierList..... | 102 |
| 3.1.6.11.1.5 | DirectoryServerType | 102 |
| 3.1.6.11.1.6 | Clustered | 103 |
| 3.1.6.11.2 | Queue..... | 103 |
| 3.1.6.11.2.1 | Pathname | 105 |
| 3.1.6.11.2.2 | QualifiedPathname | 106 |
| 3.1.6.11.2.3 | PrivacyLevel..... | 106 |
| 3.1.6.11.3 | Enterprise | 106 |
| 3.1.6.11.3.1 | Name | 107 |
| 3.1.6.11.3.2 | WeakenedSecurity | 107 |
| 3.1.6.11.3.3 | NonLDAPCapableQueueManagerNotification | 108 |
| 3.1.6.11.4 | Site | 108 |
| 3.1.6.11.5 | RoutingLink | 109 |
| 3.1.6.11.5.1 | Site1Identifier | 110 |
| 3.1.6.11.5.2 | Site2Identifier | 110 |
| 3.1.6.11.5.3 | SiteGateIdentifierList | 111 |
| 3.1.6.11.6 | User | 111 |
| 3.1.6.12 | Create Object Using LDAP | 112 |
| 3.1.6.13 | Delete Object Using LDAP | 115 |
| 3.1.6.14 | Get Object Properties Using LDAP | 117 |

| | | |
|------------|---|------------|
| 3.1.6.15 | Search Using LDAP..... | 119 |
| 3.1.6.16 | Set Object Properties Using LDAP | 121 |
| 3.1.6.17 | Set Object Security Using LDAP | 123 |
| 3.1.6.18 | Prepare an LDAP Connection | 126 |
| 3.1.6.19 | Find Object By GUID Using LDAP | 127 |
| 3.1.6.20 | Data Element Directory Attribute Tables | 128 |
| 3.1.6.20.1 | QueueManager..... | 128 |
| 3.1.6.20.2 | Queue..... | 130 |
| 3.1.6.20.3 | Enterprise | 130 |
| 3.1.6.20.4 | Site | 131 |
| 3.1.6.20.5 | RoutingLink | 131 |
| 3.1.6.20.6 | User | 132 |
| 3.1.6.21 | Shut Down an LDAP Connection..... | 132 |
| 4 | Algorithm Examples | 133 |
| 5 | Security..... | 134 |
| 5.1 | Security Considerations for Implementers..... | 134 |
| 5.1.1 | QueueManager..... | 134 |
| 5.1.2 | Queue..... | 135 |
| 5.1.3 | Enterprise | 135 |
| 5.1.4 | Site | 135 |
| 5.1.5 | RoutingLink | 136 |
| 5.1.6 | User | 136 |
| 5.1.7 | Queue Alias | 136 |
| 5.1.8 | Distribution List..... | 136 |
| 5.2 | Index of Security Parameters | 136 |
| 6 | Appendix A: Product Behavior | 137 |
| 7 | Change Tracking..... | 140 |
| 8 | Index | 141 |

1 Introduction

This document specifies the Message Queuing (MSMQ): Directory Service Schema Mapping.

[[MS-MQDMPR](#)] section 3.1.1 specifies a common abstract data model (ADM) used by all protocols in the MSMQ family. A subset of the ADM elements and ADM element attributes specified there can be stored in **Active Directory**, which provides a **Lightweight Directory Access Protocol (LDAP)** interface. The Directory Service Schema Mapping specifies an algorithm by which ADM elements are persisted as specific objects in Active Directory. It also provides a set of events that trigger LDAP operations to access those objects in Active Directory.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in RFC 2119. Sections 1.5 and 1.9 are also normative but cannot contain those terms. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are defined in [[MS-GLOS](#)]:

Active Directory
Active Directory object
Active Directory schema
discretionary access control list (DACL)
distinguished name (DN) (4)
globally unique identifier (GUID)
Lightweight Directory Access Protocol (LDAP)
security identifier (SID)
Unicode

The following terms are defined in [[MS-MQMQ](#)]:

path name
queue manager

The following terms are specific to this document:

directory string: A string as specified in [[RFC2252](#)] section 6.10.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [[RFC2119](#)]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

A reference marked "(Archived)" means that the reference document was either retired and is no longer being maintained or was replaced with a new document that provides current implementation details. We archive our documents online [[Windows Protocol](#)].

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

- [MS-ADA1] Microsoft Corporation, "[Active Directory Schema Attributes A-L](#)".
- [MS-ADA2] Microsoft Corporation, "[Active Directory Schema Attributes M](#)".
- [MS-ADA3] Microsoft Corporation, "[Active Directory Schema Attributes N-Z](#)".
- [MS-ADSC] Microsoft Corporation, "[Active Directory Schema Classes](#)".
- [MS-ADSO] Microsoft Corporation, "[Active Directory System Overview](#)".
- [MS-ADTS] Microsoft Corporation, "[Active Directory Technical Specification](#)".
- [MS-DTYP] Microsoft Corporation, "[Windows Data Types](#)".
- [MS-MQDMPR] Microsoft Corporation, "[Message Queuing \(MSMQ\): Common Data Model and Processing Rules](#)".
- [MS-MQDS] Microsoft Corporation, "[Message Queuing \(MSMQ\): Directory Service Protocol](#)".
- [MS-MQMQ] Microsoft Corporation, "[Message Queuing \(MSMQ\): Data Structures](#)".
- [MS-MQOD] Microsoft Corporation, "[Message Queuing Protocols Overview](#)".
- [RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992, <http://www.ietf.org/rfc/rfc1321.txt>
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/rfc/rfc2119.txt>
- [RFC2251] Wahl, M., Howes, T., and Kille, S., "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997, <http://www.ietf.org/rfc/rfc2251.txt>
- [RFC2252] Wahl, M., Coulbeck, A., Howes, T., and Kille, S., "Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions", RFC 2252, December 1997, <http://www.ietf.org/rfc/rfc2252.txt>

1.2.2 Informative References

[LDAP] Microsoft Corporation, "About Lightweight Directory Access Protocol", <http://msdn.microsoft.com/en-us/library/aa366075.aspx>

If you have any trouble finding [LDAP], please check [here](#).

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)".

1.3 Overview

The Message Queuing (MSMQ): Directory Service Schema Mapping is used by any protocol that manipulates the subset of the ADM elements and ADM attributes specified in [\[MS-MQDMPR\]](#) section 3.1 that can be stored in a directory, in the case in which the directory service provider is Active Directory. This algorithm, when combined with the common ADM and an understanding of the Active

Directory LDAP interface, as defined in [\[MS-ADTS\]](#), allows an abstract operation on ADM elements and ADM attributes to be reduced to a concrete LDAP operation on concrete **Active Directory objects** and attributes.

The algorithm provides access to stateful information, but it is up to Active Directory to maintain that state.

1.4 Relationship to Other Protocols

The MSMQ Directory Service Schema Mapping relies upon the LDAP interface of Active Directory, as specified in [\[MS-ADTS\]](#); references to the underlying specifications of [\[LDAP\]](#) itself are found in that document.

The Message Queuing (MSMQ): Directory Service Schema Mapping uses abstract data model (ADM) elements specified in Message Queuing (MSMQ): Common Data Model and Processing Rules [\[MS-MQDMPR\]](#) and data structures specified in Message Queuing (MSMQ): Data Structures [\[MS-MQMQ\]](#).

The Message Queuing (MSMQ): Directory Service Schema Mapping is used by the processing rules specified in Message Queuing (MSMQ): Common Data Model and Processing Rules [\[MS-MQDMPR\]](#), as shown in the diagram in [\[MS-MQDMPR\]](#) section 1.4. Protocols shown in the diagram can call the events specified in [\[MS-MQDMPR\]](#) sections [3.1.7.1.3.1](#) and [3.1.7.1.18](#) through [3.1.7.1.24](#), and the processing rules in those sections call the events specified in this algorithm.

The Message Queuing (MSMQ): Directory Service Schema Mapping is used by the processing rules specified in Message Queuing (MSMQ): Directory Service Protocol Specification [\[MS-MQDS\]](#), as shown in the diagram in [\[MS-MQDMPR\]](#) section 1.4. Protocols shown in the diagram can call the events specified in [\[MS-MQDS\]](#) sections [3.1.6.7](#) through [3.1.6.9](#), [3.1.6.11](#) through [3.1.6.13](#), and [3.1.6.15](#), and the processing rules in those sections call the events specified in this algorithm.

1.5 Prerequisites/Preconditions

It is assumed that MSMQ is operating in an environment in which Active Directory is available and in use by MSMQ. It is further assumed that any MSMQ instance has the information required to access Active Directory via LDAP in this environment.

1.6 Applicability Statement

Message Queuing (MSMQ): Directory Service Schema Mapping is applicable for implementation in an environment in which Active Directory is available and desired to be used.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

None.

2.2 Common Data Types

The following table summarizes the types defined in this specification.

| Type | Description |
|---|--|
| DirectoryOperationResult ([MS-MQDMPR] section 3.1.1.17) | An enumeration that specifies the result of a directory operation. |

2.2.1 LDAP Distinguished Names of Directory Objects

The Directory Service Schema Mapping uses the Active Directory classes listed in the following table and MUST use these **distinguished names** in LDAP queries to access objects of these classes.

| Object type | Distinguished name |
|---|--|
| mSMQQueue ([MS-ADSC] section 2.160) | CN=<queue name>, CN=msmq, CN=<computer name>, CN=Computers, <root> |
| mSMQConfiguration ([MS-ADSC] section 2.157) | CN=msmq, CN=<computer name>, CN=Computers, <root> |
| computer ([MS-ADSC] section 2.21) | CN=<computer name>, CN=Computers, <root> |
| site ([MS-ADSC] section 2.252) | CN=<site name>, CN=Sites, CN=Configuration, <root> |
| mSMQEnterpriseSettings ([MS-ADSC] section 2.158) | CN=MsmqServices, CN=Services, CN=Configuration, <root> |
| user ([MS-ADSC] section 2.263) | CN=<name>, CN=Users, <root> |
| mSMQSiteLink ([MS-ADSC] section 2.162) | CN=<routing link name>, CN=MsmqServices, CN=Services, CN=Configuration, <root> |
| mSMQSettings ([MS-ADSC] section 2.161) | CN=MSMQ Settings, CN=<computer name>, CN=Servers, CN=<site name>, CN=Sites, CN=Configuration, <root> |
| mSMQ-Custom-Recipient ([MS-ADSC] section 2.155) | CN=<name>, CN=Users, <root> |
| group ([MS-ADSC] section 2.53) | CN=<name>, CN=Users, <root> |

<queue name> MUST be the "QueueName" portion of an MSMQ Queue Name as specified in [\[MS-MQMQ\]](#) section 2.1.1.

<computer name> MUST be the "Computer" portion of an MSMQ Queue Name as specified in [\[MS-MQMQ\]](#) section 2.1.1.

<site name> MUST be the **Site.Name** ADM element attribute specified in [\[MS-MQDMPR\]](#) section 3.1.1.7.

<routing link name> is specified in section [3.1.6.1.4.2](#) of this document.

<name> is a string identifier that MUST be unique among all objects of the same type in Active Directory.

In each case, <root> MUST be a common root for these entries, which is the rootDomainNamingContext as specified in [\[MS-ADTS\]](#) section 3.1.1.3.2.16.

2.2.2 Attributes of Directory Objects

This table lists the attributes used by the Directory Service Schema Mapping for each of the Active Directory classes listed in section [2.2.1](#).

| Object | Attributes |
|-------------------|---|
| mSMQQueue | objectGUID ([MS-ADA3] section 2.44) mSMQLabelEx ([MS-ADA2] section 2.523) whenCreated ([MS-ADA3] section 2.371) whenChanged ([MS-ADA3] section 2.370) mSMQQueueType ([MS-ADA2] section 2.538) mSMQJournal ([MS-ADA2] section 2.520) mSMQQueueQuota ([MS-ADA2] section 2.537) mSMQQueueJournalQuota ([MS-ADA2] section 2.535) mSMQAuthenticate ([MS-ADA2] section 2.503) mSMQPrivacyLevel ([MS-ADA2] section 2.533) mSMQTransactional ([MS-ADA2] section 2.556) MSMQ-MulticastAddress ([MS-ADA2] section 2.500) nTSecurityDescriptor ([MS-ADA3] section 2.37) mSMQBasePriority ([MS-ADA2] section 2.504) mSMQQueueNameExt ([MS-ADA2] section 2.536) distinguishedName ([MS-ADA1] section 2.177) |
| mSMQConfiguration | objectGUID ([MS-ADA3] section 2.44) whenCreated ([MS-ADA3] section 2.371) whenChanged ([MS-ADA3] section 2.370) mSMQServiceType ([MS-ADA2] section 2.543) mSMQQuota ([MS-ADA2] section 2.539) mSMQJournalQuota ([MS-ADA2] section 2.521) mSMQForeign ([MS-ADA2] section 2.516) distinguishedName ([MS-ADA1] section 2.177) mSMQRoutingServices ([MS-ADA2] section 2.541) mSMQDsServices ([MS-ADA2] section 2.542) mSMQDependentClientServices ([MS-ADA2] section 2.510) mSMQEncryptKey ([MS-ADA2] section 2.515) nTSecurityDescriptor ([MS-ADA3] section 2.37) mSMQSites ([MS-ADA2] section 2.555) mSMQOutRoutingServers ([MS-ADA2] section 2.530) mSMQInRoutingServers ([MS-ADA2] section 2.517) mSMQComputerTypeEx ([MS-ADA2] section 2.506) |

| Object | Attributes |
|------------------------|--|
| | mSMQOSType ([MS-ADA2] section 2.529) |
| computer | mSMQSignCertificates ([MS-ADA2] section 2.544) servicePrincipalName ([MS-ADA3] section 2.253) objectSid ([MS-ADA3] section 2.45) dNSHostName ([MS-ADA1] section 2.185) operatingSystemVersion ([MS-ADA3] section 2.56) mSMQSignCertificatesMig ([MS-ADA2] section 2.545) <1> mSMQDigestsMig ([MS-ADA2] section 2.512) <2> |
| site | objectGUID ([MS-ADA3] section 2.44) cn ([MS-ADA1] section 2.110) mSMQInterval1 ([MS-ADA2] section 2.518) mSMQInterval2 ([MS-ADA2] section 2.519) distinguishedName ([MS-ADA1] section 2.177) mSMQSiteForeign ([MS-ADA2] section 2.549) nTSecurityDescriptor ([MS-ADA3] section 2.37) mSMQnt4Stub ([MS-ADA2] section 2.528) |
| mSMQEnterpriseSettings | objectGUID ([MS-ADA3] section 2.44) mSMQNameStyle ([MS-ADA2] section 2.526) mSMQCSPName ([MS-ADA2] section 2.508) mSMQLongLived ([MS-ADA2] section 2.524) mSMQVersion ([MS-ADA2] section 2.558) nTSecurityDescriptor ([MS-ADA3] section 2.37) |
| user | objectGUID ([MS-ADA3] section 2.44) distinguishedName ([MS-ADA1] section 2.177) objectSid ([MS-ADA3] section 2.45) mSMQSignCertificates ([MS-ADA2] section 2.544) mSMQDigests ([MS-ADA2] section 2.511) mSMQSignCertificatesMig ([MS-ADA2] section 2.545) <3> mSMQDigestsMig ([MS-ADA2] section 2.512) <4> |
| mSMQSiteLink | objectGUID ([MS-ADA3] section 2.44) description ([MS-ADA1] section 2.153) distinguishedName ([MS-ADA1] section 2.177) mSMQCost ([MS-ADA2] section 2.507) mSMQSite1 ([MS-ADA2] section 2.547) mSMQSite2 ([MS-ADA2] section 2.548) mSMQSiteGates ([MS-ADA2] section 2.550) mSMQSiteGatesMig ([MS-ADA2] section 2.551) <5> |
| mSMQSettings | mSMQQMID ([MS-ADA2] section 2.534) mSMQServices ([MS-ADA2] section 2.542) mSMQRoutingService ([MS-ADA2] section 2.540) mSMQDsService ([MS-ADA2] section 2.513) |

| Object | Attributes |
|-----------------------|--|
| | mSMQDependentClientService ([MS-ADA2] section 2.509) mSMQMigrated ([MS-ADA2] section 2.525) <6> |
| mSMQ-Custom-Recipient | msMQ-Recipient-FormatName ([MS-ADA2] section 2.501) objectGUID ([MS-ADA3] section 2.44) |
| group | objectGUID ([MS-ADA3] section 2.44) member ([MS-ADA2] section 2.43) |

2.2.3 mSMQEnterpriseSettings Object

As specified in [MS-ADSC] section 2.158, there MUST NOT be more than one mSMQEnterpriseSettings object in a rootDomainNamingContext ([MS-ADTS] section 3.1.1.3.2.16). There SHOULD <7> always be exactly one mSMQEnterpriseSettings object in a rootDomainNamingContext.

2.2.4 Unused Active Directory Properties

The following directory properties are present in the MSMQ's Active Directory schema ([MS-ADA2] sections 2.531, 2.552, 2.550, 2.502, 2.505, 2.522, and 2.553) but are not used in any Windows version:

- mSMQOwnerID, [MS-ADA2] section 2.531.
- mSMQSiteID, [MS-ADA2] section 2.552.
- mSMQPrevSiteGates, [MS-ADA2] section 2.550.
- MSMQ-SecuredSource, [MS-ADA2] section 2.502.
- mSMQComputerType, [MS-ADA2] section 2.505 (has been replaced with mSMQComputerTypeEx, [MS-ADA2] section 2.506).
- mSMQLabel, [MS-ADA2] section 2.522 (has been replaced with mSMQLabelEx, [MS-ADA2] section 2.523).
- mSMQSiteName, [MS-ADA2] section 2.553 (has been replaced with mSMQSiteNameEx, [MS-ADA2] section 2.554).

2.2.5 Hash String Calculation From Queue Name

The QueueName part of a **path name**, as specified in [MS-MQMQ] section 2.1.1, can exceed 64 **Unicode** characters in length. If this occurs, the QueueName must be split and the overflow stored in a second directory attribute. However, the value of the directory attribute that holds the first 64 characters is still required to be unique among all queues hosted by one **queue manager**, which might not be true if one queue manager hosts multiple queues having long names that differ only by a number at the end, for example. To enforce uniqueness in the first 64 characters, longer names are split at a lower character position, and a hash string computed according to the following algorithm is appended to the first part. See section 3.1.6.1.2.5 for more information on when this is used.

1. Let *QName* be a Unicode string initialized to the queue name for which the hash value is calculated.

2. Let *HashValue* be a 32-bit integer, initialized to zero.
3. For each Unicode character in *QName*, these steps MUST be followed:
 1. The character MUST be converted to lowercase.
 2. Let *DownShifted* be a 32-bit integer, initialized to the value of *HashValue* shifted right by eight bits.
 3. The most significant eight bits of the current character MUST be bitwise eXclusive-OR (XORed) with the least significant eight bits of *HashValue*, and the result MUST be used as an index into the table at the end of this section.
 4. *HashValue* MUST be set to the value of *DownShifted* bitwise eXclusive-OR (XORed) with the table value.
 5. *DownShifted* MUST be set to the value of *HashValue* shifted right by eight bits.
 6. The least significant eight bits of the current character MUST be bitwise eXclusive-OR (XORed) with the least significant eight bits of *HashValue*, and the result MUST be used as an index into the table at the end of this section.
 7. *HashValue* MUST be set to the value of *DownShifted* bitwise eXclusive-OR (XORed) with the table value.
4. The hash string MUST be an eight-character hexadecimal Unicode string representation of the final value of *HashValue*, with the hexadecimal digits 'a' through 'f' represented by lowercase characters.

| Index | Table value |
|-------|-------------|
| 0 | 0x00000000 |
| 1 | 0x82E0FE45 |
| 2 | 0x3302DCCD |
| 3 | 0xB1E22288 |
| 4 | 0x6605B99A |
| 5 | 0xE4E547DF |
| 6 | 0x55076557 |
| 7 | 0xD7E79B12 |
| 8 | 0xCC0B7334 |
| 9 | 0x4EEB8D71 |
| 10 | 0xFF09AFF9 |
| 11 | 0x7DE951BC |
| 12 | 0xAA0ECAAE |
| 13 | 0x28EE34EB |

| Index | Table value |
|--------------|--------------------|
| 14 | 0x990C1663 |
| 15 | 0x1BECE826 |
| 16 | 0xAED5C62F |
| 17 | 0x2C35386A |
| 18 | 0x9DD71AE2 |
| 19 | 0x1F37E4A7 |
| 20 | 0xC8D07FB5 |
| 21 | 0x4A3081F0 |
| 22 | 0xFBD2A378 |
| 23 | 0x79325D3D |
| 24 | 0x62DEB51B |
| 25 | 0xE03E4B5E |
| 26 | 0x51DC69D6 |
| 27 | 0xD33C9793 |
| 28 | 0x04DB0C81 |
| 29 | 0x863BF2C4 |
| 30 | 0x37D9D04C |
| 31 | 0xB5392E09 |
| 32 | 0x6B68AC19 |
| 33 | 0xE988525C |
| 34 | 0x586A70D4 |
| 35 | 0xDA8A8E91 |
| 36 | 0x0D6D1583 |
| 37 | 0x8F8DEBC6 |
| 38 | 0x3E6FC94E |
| 39 | 0xBC8F370B |
| 40 | 0xA763DF2D |
| 41 | 0x25832168 |
| 42 | 0x946103E0 |
| 43 | 0x1681FDA5 |

| Index | Table value |
|--------------|--------------------|
| 44 | 0xC16666B7 |
| 45 | 0x438698F2 |
| 46 | 0xF264BA7A |
| 47 | 0x7084443F |
| 48 | 0xC5BD6A36 |
| 49 | 0x475D9473 |
| 50 | 0xF6BF66FB |
| 51 | 0x745F48BE |
| 52 | 0xA3B8D3AC |
| 53 | 0x21582DE9 |
| 54 | 0x90BA0F61 |
| 55 | 0x125AF124 |
| 56 | 0x09B61902 |
| 57 | 0x8B56E747 |
| 58 | 0x3AB4C5CF |
| 59 | 0xB8543B8A |
| 60 | 0x6FB3A098 |
| 61 | 0xED535EDD |
| 62 | 0x5CB17C55 |
| 63 | 0xDE518210 |
| 64 | 0xD6D15832 |
| 65 | 0x5431A677 |
| 66 | 0xE5D384FF |
| 67 | 0x67337ABA |
| 68 | 0xB0D4E1A8 |
| 69 | 0x32341FED |
| 70 | 0x83D63D65 |
| 71 | 0x0136C320 |
| 72 | 0x1ADA2B06 |
| 73 | 0x983AD543 |

| Index | Table value |
|--------------|--------------------|
| 74 | 0x29D8F7CB |
| 75 | 0xAB38098E |
| 76 | 0x7CDF929C |
| 77 | 0xFE3F6CD9 |
| 78 | 0x4FDD4E51 |
| 79 | 0xCD3DB014 |
| 80 | 0x78049E1D |
| 81 | 0xFAE46058 |
| 82 | 0x4B0642D0 |
| 83 | 0xC9E6BC95 |
| 84 | 0x1E012787 |
| 85 | 0x9CE1D9C2 |
| 86 | 0x2D03FB4A |
| 87 | 0xAF3050F |
| 88 | 0xB40FED29 |
| 89 | 0x36EF136C |
| 90 | 0x870D31E4 |
| 91 | 0x05EDCFA1 |
| 92 | 0xD20A54B3 |
| 93 | 0x50EAAAF6 |
| 94 | 0xE108887E |
| 95 | 0x63E8763B |
| 96 | 0xBDB9F42B |
| 97 | 0x3F590A6E |
| 98 | 0x8EBB28E6 |
| 99 | 0x0C5BD6A3 |
| 100 | 0xDBBC4DB1 |
| 101 | 0x595CB3F4 |
| 102 | 0xE8BE917C |
| 103 | 0x6A5E6F39 |

| Index | Table value |
|--------------|--------------------|
| 104 | 0x71B2871F |
| 105 | 0xF352795A |
| 106 | 0x42B05BD2 |
| 107 | 0xC050A597 |
| 108 | 0x17B73E85 |
| 109 | 0x9557C0C0 |
| 110 | 0x24B5E248 |
| 111 | 0xA6551C0D |
| 112 | 0x136C3204 |
| 113 | 0x918CCC41 |
| 114 | 0x206EEEC9 |
| 115 | 0xA28E108C |
| 116 | 0x75698B9E |
| 117 | 0xF78975DB |
| 118 | 0x466B5753 |
| 119 | 0xC48BA916 |
| 120 | 0xDF674130 |
| 121 | 0x5D87BF75 |
| 122 | 0xEC659DFD |
| 123 | 0x6E8563B8 |
| 124 | 0xB962F8AA |
| 125 | 0x3B8206EF |
| 126 | 0x8A602467 |
| 127 | 0x0880DA22 |
| 128 | 0x9B619023 |
| 129 | 0x19816E66 |
| 130 | 0xA8634CEE |
| 131 | 0x2A83B2AB |
| 132 | 0xFD6429B9 |
| 133 | 0x7F84D7FC |

| Index | Table value |
|--------------|--------------------|
| 134 | 0xCE66F574 |
| 135 | 0x4C860B31 |
| 136 | 0x576AE317 |
| 137 | 0xD58A1D52 |
| 138 | 0x64683FDA |
| 139 | 0xE688C19F |
| 140 | 0x316F5A8D |
| 141 | 0xB38FA4C8 |
| 142 | 0x026D8640 |
| 143 | 0x808D7805 |
| 144 | 0x35B4560C |
| 145 | 0xB754A849 |
| 146 | 0x06B68AC1 |
| 147 | 0x84567484 |
| 148 | 0x53B1EF96 |
| 149 | 0xD15111D3 |
| 150 | 0x60B3335B |
| 151 | 0xE253CD1E |
| 152 | 0xF9BF2538 |
| 153 | 0x7B5FDB7D |
| 154 | 0xCABDF9F5 |
| 155 | 0x485D07B0 |
| 156 | 0x9FBA9CA2 |
| 157 | 0x1D5A62E7 |
| 158 | 0xACB8406F |
| 159 | 0x2E58BE2A |
| 160 | 0xF0093C3A |
| 161 | 0x72E9C27F |
| 162 | 0xC30BE0F7 |
| 163 | 0x41EB1EB2 |

| Index | Table value |
|--------------|--------------------|
| 164 | 0x960C85A0 |
| 165 | 0x14EC7BE5 |
| 166 | 0xA50E596D |
| 167 | 0x27EEA728 |
| 168 | 0x3C024F0E |
| 169 | 0xBEE2B14B |
| 170 | 0x0F0093C3 |
| 171 | 0x8DE06D86 |
| 172 | 0x5A07F694 |
| 173 | 0xD8E708D1 |
| 174 | 0x69052A59 |
| 175 | 0xEBE5D41C |
| 176 | 0x5EDCFA15 |
| 177 | 0xDC3C0450 |
| 178 | 0x6DDE26D8 |
| 179 | 0xEF3ED89D |
| 180 | 0x38D9438F |
| 181 | 0xBA39BDCA |
| 182 | 0x0BDB9F42 |
| 183 | 0x893B6107 |
| 184 | 0x92D78921 |
| 185 | 0x10377764 |
| 186 | 0xA1D555EC |
| 187 | 0x2335ABA9 |
| 188 | 0xF4D230BB |
| 189 | 0x7632CEFE |
| 190 | 0xC7D0EC76 |
| 191 | 0x45301233 |
| 192 | 0x4DB0C811 |
| 193 | 0xCF503654 |

| Index | Table value |
|--------------|--------------------|
| 194 | 0x7EB214DC |
| 195 | 0xFC52EA99 |
| 196 | 0x2BB5718B |
| 197 | 0xA9558FCE |
| 198 | 0x18B7AD46 |
| 199 | 0x9A575303 |
| 200 | 0x81BBBB25 |
| 201 | 0x035B4560 |
| 202 | 0xB2B967E8 |
| 203 | 0x305999AD |
| 204 | 0xE7BE02BF |
| 205 | 0x655EFCFA |
| 206 | 0xD4BCDE72 |
| 207 | 0x565C2037 |
| 208 | 0xE3650E3E |
| 209 | 0x6185F07B |
| 210 | 0xD067D2F3 |
| 211 | 0x52872CB6 |
| 212 | 0x8560B7A4 |
| 213 | 0x078049E1 |
| 214 | 0xB6626B69 |
| 215 | 0x3482952C |
| 216 | 0x2F6E7D0A |
| 217 | 0xAD8E834F |
| 218 | 0x1C6CA1C7 |
| 219 | 0x9E8C5F82 |
| 220 | 0x496BC490 |
| 221 | 0xCB8B3AD5 |
| 222 | 0x7A69185D |
| 223 | 0xF889E618 |

| Index | Table value |
|--------------|--------------------|
| 224 | 0x26D86408 |
| 225 | 0xA4389A4D |
| 226 | 0x15DAB8C5 |
| 227 | 0x973A4680 |
| 228 | 0x40DDDD92 |
| 229 | 0xC23D23D7 |
| 230 | 0x73DF015F |
| 231 | 0xF13FFF1A |
| 232 | 0xEAD3173C |
| 233 | 0x6833E979 |
| 234 | 0xD9D1CBF1 |
| 235 | 0x5B3135B4 |
| 236 | 0x8CD6AEA6 |
| 237 | 0x0E3650E3 |
| 238 | 0xBF4726B |
| 239 | 0x3D348C2E |
| 240 | 0x880DA227 |
| 241 | 0x0AED5C62 |
| 242 | 0xBB0F7EEA |
| 243 | 0x39EF80AF |
| 244 | 0xEE081BBD |
| 245 | 0x6CE8E5F8 |
| 246 | 0xDD0AC770 |
| 247 | 0x5FEA3935 |
| 248 | 0x4406D113 |
| 249 | 0xC6E62F56 |
| 250 | 0x77040DDE |
| 251 | 0xF5E4F39B |
| 252 | 0x22036889 |
| 253 | 0xA0E396CC |

| Index | Table value |
|-------|-------------|
| 254 | 0x1101B444 |
| 255 | 0x93E14A01 |

2.2.6 LDAP Result Code to DirectoryOperationResult Mapping

The algorithm converts an LDAP result code *TaskReturnStatus* to a **DirectoryOperationResult** according to the following table.

| LDAP result code | DirectoryOperationResult enumeration |
|----------------------------------|--------------------------------------|
| Success (0) | Success |
| noSuchObject (32) | ObjectNotFound |
| noSuchAttribute (16) | AttributeNotFound |
| entryAlreadyExists (68) | ObjectAlreadyExists |
| Other codes not listed preceding | GenericError |

2.3 Queue Alias

A queue alias is a directory object of the mSMQ-Custom-Recipient class type that associates a directory service path and a user-defined string (alias) with a direct, public, or private format name (see [\[MS-MQMQ\]](#) sections [2.1.2](#), [2.1.3](#), and [2.1.4](#)). <8> MSMQ instances MUST use the distinguished name (DN) for objects of class mSMQ-Custom-Recipient as specified in section [2.2.1](#) in LDAP queries to access the objects and retrieve the format name of a queue from the msMQ-Recipient-FormatName attribute of the queue alias object.

2.4 Directory Service Schema Elements

The algorithm accesses the following Directory Service schema classes and attributes listed in the following table(s).

For the syntactic specifications of the following <Class> or <Class><Attribute> pairs, refer to Active Directory Domain Services (AD/DS) ([\[MS-ADA1\]](#), [\[MS-ADA2\]](#), [\[MS-ADA3\]](#)).

| Class | Attribute |
|-----------|--|
| mSMQQueue | distinguishedName mSMQAuthenticate mSMQBasePriority mSMQJournal mSMQLabelEx MSMQ-MulticastAddress mSMQPrivacyLevel mSMQQueueJournalQuota mSMQQueueNameExt mSMQQueueQuota mSMQQueueType |

| Class | Attribute |
|------------------------|--|
| | mSMQTransactional objectGUID nTSecurityDescriptor whenChanged whenCreated |
| mSMQConfiguration | distinguishedName mSMQComputerTypeEx mSMQDependentClientServices mSMQDsServices mSMQEncryptKey mSMQForeign mSMQInRoutingServers mSMQJournalQuota mSMQOSType mSMQOutRoutingServers mSMQQuota mSMQRoutingServices mSMQServiceType mSMQSites nTSecurityDescriptor objectGUID whenChanged whenCreated |
| computer | dNSHostName mSMQSignCertificates objectSid servicePrincipalName operatingSystemVersion |
| site | cn distinguishedName mSMQInterval1 mSMQInterval2 mSMQnt4Stub mSMQSiteForeign nTSecurityDescriptor objectGUID |
| mSMQEnterpriseSettings | mSMQCSPName mSMQLongLived mSMQNameStyle mSMQVersion nTSecurityDescriptor objectGUID |

| Class | Attribute |
|--------------|---|
| user | distinguishedName mSMQDigests mSMQSignCertificates objectGUID objectSid |
| mSMQSiteLink | description distinguishedName mSMQCost mSMQSite1 mSMQSite2 mSMQSiteGates objectGUID |
| mSMQSettings | mSMQDependentClientService mSMQDsService mSMQMID mSMQRoutingService mSMQServices |

3 Protocol Details

3.1 Algorithm Details

This algorithm is specified in terms of a set of abstract local events that are raised by the processing rules within the common processing rules specified in [\[MS-MQDMPR\]](#). The algorithm converts these events to LDAP operations against Active Directory. Three operations ([Read Directory Begin \(section 3.1.6.4\)](#), [Read Directory Next \(section 3.1.6.5\)](#), and [Read Directory End \(section 3.1.6.6\)](#)) have shared state; the rest are independent. The processing rules of the abstract events describe the conversion of arguments from abstract to concrete, which can involve some simple computation; the results returned by the transport are returned after conversion of results from concrete to abstract, which again can involve some simple computation.

3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this algorithm. The specified organization is provided to facilitate the explanation of how the algorithm behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that specified in this document.

The abstract data model for this algorithm comprises elements that are private to this algorithm and consists of elements that are specified in this algorithm and elements that are specified in [\[MS-MQDMPR\]](#) section 3.1.1. Although the abstract data model for this algorithm contains elements that are specified in [\[MS-MQDMPR\]](#), this algorithm does not share instances of any of these elements or state with any protocol. The relationship between this algorithm, a queue manager, and MSMQ protocols is described in [\[MS-MQOD\]](#).

Section [3.1.1.1](#) details the elements from the ADM that are specified in [\[MS-MQDMPR\]](#). Sections [3.1.1.2](#) and [3.1.1.3](#) detail the ADM elements that are specific to this algorithm. Section [3.1.1.4](#) details an extension to an ADM element specified in [\[MS-MQDMPR\]](#).

3.1.1.1 Externally Defined Data Elements

This algorithm accesses the following externally defined ADM elements:

- **DirectoryObject:** [\[MS-MQDMPR\]](#) section 3.1.1.
- **QueueManager:** [\[MS-MQDMPR\]](#) section 3.1.1.1.
- **Queue:** [\[MS-MQDMPR\]](#) section 3.1.1.2.
- **Enterprise:** [\[MS-MQDMPR\]](#) section 3.1.1.6.
- **Site:** [\[MS-MQDMPR\]](#) section 3.1.1.7.
- **RoutingLink:** [\[MS-MQDMPR\]](#) section 3.1.1.8.
- **User:** [\[MS-MQDMPR\]](#) section 3.1.1.15.

3.1.1.2 ReadDirectoryIteratorState Data Element

The **ReadDirectoryIteratorState** ADM element represents the state associated with a directory read initiated by a [Read Directory Begin \(section 3.1.6.4\)](#) event. This ADM element MUST contain the following ADM attributes:

- **Handle:** A **HANDLE** ([\[MS-DTYP\]](#) section 2.2.16) that identifies the **ReadDirectoryIteratorState** ADM element instance. This identifier MUST NOT change over the lifetime of the **ReadDirectoryIteratorState** ADM element instance.
- **DataElementType:** A string that is the name of a sub-type of a **DirectoryObject** ADM element.
- **LDAPAttributeList:** The names of the Active Directory attributes to be read from the directory.
- **LDAPState:** A list of sublists, where each sublist consists of the values returned by an LDAP search for the attributes of one Active Directory object, in the same order as the attribute names in the **LDAPAttributeList** ADM attribute.
- **AttributeList:** The names of the attributes to be read from the directory.

3.1.1.3 ReadDirectoryIteratorStateCollection Data Element

The **ReadDirectoryIteratorStateCollection** ADM element is a collection that MUST contain a reference to every existing **ReadDirectoryIteratorState** ADM element instance.

3.1.1.4 User Data Element

The algorithm MUST maintain private state for each **User** ADM element instance in addition to the state specified for the **User** ADM element in [\[MS-MQDMPR\]](#) section 3.1.1.15. The following additional ADM attribute is used to reference this private state:

- **FullPath:** A distinguished name that can be used to look up in Active Directory the user directory object corresponding to this **User** ADM element instance. This is a directory attribute, as specified in [\[MS-MQDMPR\]](#) section 3.1.1.

3.1.1.5 CachedConfigurationNamingContext

The **CachedConfigurationNamingContext** ADM element is a string that is the distinguished name of the root of the configuration tree in Active Directory. The value is obtained from Active Directory at algorithm initialization time, as specified in section [3.1.3](#).

3.1.1.6 CachedLDAPConnection

This ADM element represents an LDAP connection that can be reused if multiple directory operations occur simultaneously and defines the following ADM attributes:

Handle: Either an **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2) that is connected to an Active Directory server and that is ready to perform LDAP operations or NULL.

RefCount: An integer that represents the count of directory operations using the connection represented by the **Handle** ADM attribute. This count is incremented when an operation starts using the connection, as specified in section [3.1.6.18](#), and is decremented when an operation finishes using the connection, as specified in section [3.1.6.21](#).

3.1.2 Timers

The Directory Service Schema Mapping algorithm does not use timers.

3.1.3 Initialization

The algorithm MUST perform these initialization tasks before any directory operation (sections [3.1.6.1](#) through [3.1.6.9](#)) is called:

- Set the **CachedLDAPConnection.Handle** (section [3.1.1.6](#)) ADM element attribute to NULL and the **CachedLDAPConnection.RefCount** ADM element attribute to zero.
- Raise a [Prepare an LDAP Connection \(section 3.1.6.18\)](#) event. The event takes no arguments. If the *rStatus* returned is not **DirectoryOperationResult.Success**, the **CachedConfigurationNamingContext** ADM element MUST be set to an empty string, and processing MUST end. Otherwise, let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2) initialized to the value returned in *rADConnection*.
- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - *messageID* = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - *protocolOp* = *searchRequest*
 - *controls* = none
 - *baseObject* = an empty string
 - *scope* = *baseObject*
 - *derefAliases* = *neverDerefAliases*
 - *sizeLimit* = 0
 - *timeLimit* = 0
 - *typesOnly* = FALSE
 - *filter* = "(objectClass=*)"
 - *attributes* = an empty list
- Perform the [Performing an LDAP Operation on an ADConnection](#) task ([\[MS-ADSO\]](#) section 6.2.6.1.6) with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the **CachedConfigurationNamingContext** ADM element MUST be set to an empty string. If the result message returned in *TaskOutputResultMessages* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the **CachedConfigurationNamingContext** ADM element MUST be set to an empty string. Otherwise, the **CachedConfigurationNamingContext** ADM element MUST be set to the value of the *configurationNamingContext* attribute extracted from the result message returned in *TaskOutputResultMessages*.
- Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.

3.1.4 Message Processing Events and Sequencing Rules

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

3.1.6.1 Create Directory Object

This event MUST be generated with the following arguments:

- *iDirectoryObject*: A **DirectoryObject** ADM element instance to be created in the directory.
- *iAttributeList* (Optional): An array of strings containing the subset of the names of the directory attributes of the *iDirectoryObject* argument to be persisted in the directory. If this argument is not supplied, all directory attributes of the *iDirectoryObject* argument that can be persisted at create time are persisted. The following sections specify the individual types list attributes that can be persisted at create time.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.
- *rObjectGUID*: The **GUID** of the newly created **DirectoryObject** ADM element instance. This value is undefined if *rStatus* does not equal **DirectoryOperationResult.Success**.

The MQDSSM algorithm MUST perform the following actions to process this event:

- Determine the type of the *iDirectoryObject* argument. If the type is not one of **QueueManager**, **Queue**, **Site**, or **RoutingLink**, *rStatus* MUST be set to the **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing MUST end.
- The create operation logic for the type of ADM element specified by the *iDirectoryObject* argument MUST be performed as specified in the sections listed following. The *rStatus* and *rObjectGUID* values specified in each section MUST be returned.
 - [QueueManager \(section 3.1.6.1.1\)](#)
 - [Queue \(section 3.1.6.1.2\)](#)
 - [Site \(section 3.1.6.1.3\)](#)
 - [RoutingLink \(section 3.1.6.1.4\)](#)

3.1.6.1.1 QueueManager

3.1.6.1.1.1 Preconditions

The arguments supplied to the [Create Directory Object \(section 3.1.6.1\)](#) event MUST meet these requirements:

- If *iAttributeList* is provided, the name "SiteIdentifierList" MUST be present in the list.
- The *iDirectoryObject.SiteIdentifierList* and *iDirectoryObject.ComputerName* ADM attributes MUST be populated.
- If *iAttributeList* is provided, it MUST NOT contain the name **ConnectedNetworkIdentifierList**. This attribute is not supported in Active Directory-based environments.

If any of these conditions is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.1.1.2 Creation

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **QueueManager** ADM element attributes listed in section [3.1.6.20.1](#), except **ConnectedNetworkIdentifierList**.
2. Let *ConfigurationAttributeList* be a list of mSMQConfiguration attribute names, initialized to be empty. For each mSMQConfiguration attribute listed in the following table, if the corresponding **QueueManager** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the mSMQConfiguration attribute name to *ConfigurationAttributeList* and compute the value for the attribute as shown. **QueueManager** ADM element attribute names that appear in *iAttributeList* but that do not appear in the following table or in subsections referenced by the table MUST be ignored.

| mSMQConfiguration attribute | Attribute value computation |
|--|--|
| mSMQComputerTypeEx ([MS-ADA2] section 2.506) | Directory string transformed from <i>iDirectoryObject.QueueManagerVersion</i> . |
| mSMQOSType ([MS-ADA2] section 2.529) | Computed from <i>iDirectoryObject.OperatingSystemType</i> . See section 3.1.6.1.1.4 . |
| mSMQServiceType ([MS-ADA2] section 2.543) | Computed from <i>iDirectoryObject.DirectoryServer</i> , <i>iDirectoryObject.DirectoryServerType</i> , <i>iDirectoryObject.RemoteAccessServer</i> , and <i>iDirectoryObject.RoutingServer</i> . See section 3.1.6.1.1.5 . |
| mSMQQuota ([MS-ADA2] section 2.539) | Integer copied from <i>iDirectoryObject.QueueManagerQuota</i> . |
| mSMQJournalQuota ([MS-ADA2] section 2.521) | Integer copied from <i>iDirectoryObject.JournalQuota</i> . |
| mSMQForeign ([MS-ADA2] section 2.516) | If <i>iDirectoryObject.ForeignSystem</i> is TRUE, mSMQForeign is 0x01 Else mSMQForeign is 0x00. |
| mSMQSites ([MS-ADA2] section 2.555) | List of GUIDs copied from <i>iDirectoryObject.SiteIdentifierList</i> . |
| mSMQOutRoutingServers ([MS-ADA2] section 2.530) | Computed from <i>iDirectoryObject.OutRoutingServerIdentifierList</i> . See section 3.1.6.1.1.6 . |
| mSMQInRoutingServers ([MS-ADA2] section 2.517) | Computed from <i>iDirectoryObject.InRoutingServerIdentifierList</i> . See section 3.1.6.1.1.7 . |
| mSMQRoutingServices ([MS-ADA2] section 2.541) | If <i>iDirectoryObject.RoutingServer</i> is TRUE, mSMQRoutingServices is 0x01 Else |

| mSMQConfiguration attribute | Attribute value computation |
|---|--|
| | mSMQRoutingServices is 0x00. |
| mSMQDsServices ([MS-ADA2] section 2.514) | If <i>iDirectoryObject</i> . DirectoryServer is TRUE, mSMQDsServices is 0x01 Else mSMQDsServices is 0x00. |
| mSMQDependentClientServices ([MS-ADA2] section 2.510) | If <i>iDirectoryObject</i> . SupportingServer is TRUE, mSMQDependentClientServices is 0x01 Else mSMQDependentClientServices is 0x00. |
| mSMQEncryptKey ([MS-ADA2] section 2.515) | MQDSPUBLICKEYS structure ([MS-MQMQ] section 2.2.2) copied from <i>iDirectoryObject</i> . PublicEncryptionKeyList . |
| nTSecurityDescriptor ([MS-ADA3] section 2.37) | Computed from <i>iDirectoryObject</i> . Security . See section 3.1.6.1.1.8 . |

3. A [Create Object Using LDAP \(section 3.1.6.12\)](#) event MUST be generated with the following arguments:
 - *iParentPath* := a distinguished name of the form specified for a computer object in section [2.2.1](#), where "<computer name>" is the value of *iDirectoryObject*.**ComputerName**
 - *iChildName* := "msmq"
 - *iObjectClass* := "mSMQConfiguration"
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *ConfigurationAttributeList* and the corresponding values, as computed in step 2.
4. If the Create Object Using LDAP event returns an *rStatus* value of **DirectoryOperationResult.ObjectAlreadyExists**, a [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iPath* := a distinguished name of the form specified for mSMQConfiguration in section [2.2.1](#), where "<computer name>" is the value of *iDirectoryObject*.**ComputerName**
 - *iAttributes* := a list containing the attribute name "objectGUID"

If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rObjectGUID* is undefined, and processing MUST end. Otherwise, *rStatus* is set to the first element in *rValues* returned by the Get Object Properties Using LDAP event.
5. Else if the Create Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Create Object Using LDAP event, *rObjectGUID* is undefined, and processing MUST end. Otherwise, *rObjectGuid* is set to the *rObjectGuid* returned by the Create Object Using LDAP event.

3.1.6.1.1.3 Postprocessing

1. If the [PublicSigningKeyList \(section 3.1.6.7.1.10\)](#) ADM attribute is present in *iAttributeList*, its value MUST be written to the directory:
 - A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event MUST be generated with the following arguments:
 - *iPath* := distinguished name for a computer object MUST be constructed according to the format specified in section [2.2.1](#), where "<computer name>" is the value of *iDirectoryObject.ComputerName*.
 - *iAttributeList* := a list of name-value pairs consisting of the attribute names "mSMQSignCertificates", "mSMQDigests", "mSMQSignCertificatesMig", and "mSMQDigestsMig" and the corresponding values computed as specified in section [3.1.6.1.1.9](#).
 - If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, *rObjectGUID* is undefined, and processing MUST end.
2. If one or more of the **RoutingServer**, **DirectoryServer**, or **SupportingServer** ADM attributes on *iDirectoryObject* is TRUE, one or more mSMQSettings objects MUST be created as well. For each GUID in the value of *iDirectoryObject.SiteIdentifierList*, these steps MUST be performed:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from the **SiteIdentifierList**
 - *iAttributeList* := Null
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.ObjectNotFound**:
 - The current GUID MUST be skipped.
 - Else if the *rStatus* returned by the Read Directory event is set to anything other than **DirectoryOperationResult.Success**:
 - *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing MUST end.
 - Else if the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**:
 - Generate a [Create Object Using LDAP \(section 3.1.6.12\)](#) event with the following arguments:
 - *iParentPath* := a distinguished name of the form specified for an mSMQSettings object in section [2.2.1](#) MUST be constructed, where "<computer name>" is the value of **QueueManager.ComputerName**, and "<site name>" is the value of the **Name** ADM attribute of the returned **Site** ADM element instance, then the first comma-separated element removed

- *iChildName* := "MSMQ Settings"
- *iObjectClass* := "mSMQSettings"
- *iAttributes* := a list of name-value pairs consisting of the attribute names in the following table and the corresponding values copied from *ConfigurationAttributeList*

| mSMQSettings attribute | mSMQConfiguration attribute |
|--|-----------------------------|
| mSMQQMID ([MS-ADA2] section 2.534) | objectGUID |
| mSMQServices ([MS-ADA2] section 2.542) | mSMQServiceType |
| mSMQRoutingService ([MS-ADA2] section 2.540) | mSMQRoutingServices |
| mSMQDsService ([MS-ADA2] section 2.513) | mSMQDsServices |
| mSMQDependentClientService ([MS-ADA2] section 2.509) | mSMQDependentClientServices |

- If the Create Object Using LDAP event returns an *rStatus* value of **DirectoryOperationResult.ObjectAlreadyExists**, a [Delete Object Using LDAP \(section 3.1.6.13\)](#) event MUST be generated with the *iPath* argument set to "CN=MSMQ Settings, <parent path>", where <parent path> is the *iParentPath* argument of the preceding Create Object Using LDAP event. If the *rStatus* returned by the Delete Object Using LDAP event is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Delete Object Using LDAP event, *rObjectGUID* is undefined, and processing MUST end. Otherwise, the Create Object Using LDAP event in the preceding step MUST be invoked again.
- If the Create Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, *rObjectGUID* is undefined, and processing MUST end.

3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.1.1.4 mSMQOSType

The [mSMQOSType](#) attribute is an integer that MUST have one of the values listed in the following table. The **OperatingSystemType** ADM attribute is an enumeration. The value MUST be converted according to this table.

| OperatingSystemType ADM attribute value | mSMQOSType value |
|---|------------------|
| Other | 0x00000000 |
| Foreign | 0x00000100 |
| Windows 95 | 0x00000200 |
| WinClient | 0x00000300 |
| WinServer | 0x00000400 |
| WinEnt | 0x00000500 |

3.1.6.1.1.5 mSMQServiceType

The mSMQServiceType attribute is a bitmap that MUST be computed according to this algorithm:

1. Let *FinalValue* be an integer, initialized to 0x00000000.
2. If *iDirectoryObject.RoutingServer* is populated and TRUE, the 0x00000001 bit of *FinalValue* MUST be set.
3. If *iDirectoryObject.RemoteAccessServer* is populated and TRUE, the 0x00000010 bit of *FinalValue* MUST be set.
4. If *iDirectoryObject.DirectoryServer* is populated and TRUE, and *iDirectoryObject.DirectoryServerType* is populated and does not have the enumeration value **Standalone**, one bit of *FinalValue* MUST be set according to the following table.

| DirectoryServerType value | FinalValue bit to be set |
|-----------------------------|--------------------------|
| BackupSiteController | 0x00000002 |
| PrimarySiteController | 0x00000004 |
| PrimaryEnterpriseController | 0x00000008 |

5. The value of the mSMQServiceType attribute MUST be the value of *FinalValue*.

3.1.6.1.1.6 mSMQOutRoutingServers

The value of the mSMQOutRoutingServers attribute MUST be computed according to the following algorithm:

1. Let *DNList* be a list of distinguished names, initialized to be empty.
2. For each GUID in *iDirectoryObject.OutRoutingServerIdentifierList*, these steps MUST be performed:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from *iDirectoryObject.OutRoutingServerIdentifierList*
 - *iAttributeList* := An array of the following **QueueManager** ADM element attribute names:
 - **FullPath**
 - If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, this GUID MUST be skipped.
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of the *rDirectoryObject.FullPath* returned by the Read Directory event must be added to *DNList*.

3. The value of `mSMQOutRoutingServers` MUST be the value of `DNList`.

3.1.6.1.1.7 mSMQInRoutingServers

The value of the `mSMQInRoutingServers` attribute MUST be computed according to the following algorithm:

1. Let `DNList` be a list of distinguished names, initialized to be empty.
2. For each GUID in `iDirectoryObject.InRoutingServerIdentifierList`, these steps MUST be performed:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - `iDirectoryObjectType` := "QueueManager"
 - `iFilter` := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from `iDirectoryObject.InRoutingServerIdentifierList`
 - `iAttributeList` := An array of the following **QueueManager** ADM element attribute names:
 - **FullPath**
 - If the `rStatus` returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, this GUID MUST be skipped.
 - If the `rStatus` returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of the `iDirectoryObject.FullPath` returned by the Read Directory event must be added to `DNList`.
3. The value of `mSMQInRoutingServers` MUST be the value of `DNList`.

3.1.6.1.1.8 nTSecurityDescriptor

The value of the `nTSecurityDescriptor` attribute MUST be computed according to the following algorithm:

1. Let `OwnerSid` and `MachineSid` be **Security Identifiers (SIDs)** ([\[MS-DTYP\]](#) section 2.4.2), both initialized to zero.
2. A distinguished name MUST be formed by removing the first element of the distinguished name computed in step 3 of section [3.1.6.1.1.2](#). The `objectSid` attribute of the computer object referenced by this distinguished name MUST be read via LDAP. If the read operation fails, the values of `MachineSid` and `OwnerSid` MUST be unchanged. Otherwise, the values of `MachineSid` and `OwnerSid` MUST be set to the value read.
3. Let `UserSid` be a SID. If `iDirectoryObject.Security` is populated and the **security descriptor** that is the value of `iDirectoryObject.Security` contains an owner SID, `UserSid` MUST be initialized with the value of that owner SID. Otherwise, `UserSid` MUST be initialized to the SID of the user under whose identity the current thread is running.
4. Let `WorldAccess` be an **MQQACCESSMASK** ([\[MS-MQMQ\]](#) section 2.2.23) enumerated value, which MUST be initialized to be `MQSEC_MACHINE_WORLD_RIGHTS`.

5. If the ADM attribute name **Foreign** is present in *iAttributeList* and *iDirectoryObject.Foreign* is populated and the value of *iDirectoryObject.Foreign* is TRUE, *WorldAccess* MUST be set to MQSEC_MACHINE_GENERIC_WRITE.
6. Let *FinalSecurity* be a **SECURITY_DESCRIPTOR** as specified in [\[MS-DTYP\]](#) section 2.4.6, initialized to be empty.
7. An **ACCESS_ALLOWED_ACE** ([\[MS-DTYP\]](#) section 2.4.4.2) with the **Mask** field set to *WorldAccess* and containing the well-known SID with string representation S-1-1-0 (relative identifier SECURITY_WORLD_RID combined with identifier authority SECURITY_WORLD_SID_AUTHORITY) MUST be added to the **Dacl** in *FinalSecurity*.
8. If *OwnerSid* is nonzero, an **ACCESS_ALLOWED_ACE** structure with a **Mask** field set to MQSEC_MACHINE_GENERIC_ALL and containing the SID in *OwnerSid* MUST be added to the **Dacl** in *FinalSecurity*.
9. If *MachineSid* is nonzero, an **ACCESS_ALLOWED_ACE** structure with a **Mask** set to MQSEC_MACHINE_SELF_RIGHTS and containing the SID in *MachineSid* MUST be added to the **Dacl** in *FinalSecurity*.
10. If *UserSid* is nonzero, an **ACCESS_ALLOWED_ACE** structure with a **Mask** field set to MQSEC_MACHINE_GENERIC_ALL and containing the SID in *UserSid* MUST be added to the **Dacl** in *FinalSecurity*.
11. The **SECURITY_DESCRIPTOR** structure in *FinalSecurity* MUST be converted to self-relative format (see [\[MS-DTYP\]](#) section 2.4.6).
12. The value of the nTSecurityDescriptor attribute MUST be the value of *FinalSecurity*.

3.1.6.1.1.9 mSMQSignCertificates and mSMQDigests

The **PublicSigningKeyList** attribute is stored in two attributes on the computer object. The mSMQSignCertificates ([\[MS-ADA2\]](#) section 2.544) attribute contains an **MQUERSIGNCERTS** ([\[MS-MQMQ\]](#) section 2.2.21) structure, which contains **MQUERSIGNCERT** ([\[MS-MQMQ\]](#) section 2.2.22) structures, which in turn contain individual X.509-encoded certificates. The mSMQDigests ([\[MS-ADA2\]](#) section 2.511) attribute contains an array of MD5 hashes of the certificates stored in the mSMQSignCertificates attribute, mirroring the values of the **Digest** fields of the **MQUERSIGNCERT** structures. Each array element MUST contain the 16-byte output of the MD5 algorithm, as specified in [\[RFC1321\]](#). The certificates and digests MUST be in the same order, but there is no other sorting requirement.

The values of mSMQSignCertificates and mSMQDigests MUST be computed according to the following algorithm:

If the **PublicSigningKeyList** ADM attribute name is present in *iAttributeList* and *iDirectoryObject.PublicSigningKeyList* is populated:

- Copy the bytes in *iDirectoryObject.PublicSigningKeyList*, which is an **MQUERSIGNCERTS** structure, to mSMQSignCertificates.
- For each **MQUERSIGNCERT** structure in mSMQSignCertificates:
 - Append the bytes of the **Digest** field in the structure to mSMQDigests.

The attributes mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545) and mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.512) MAY [<9>](#) be set to the values of mSMQSignCertificates and mSMQDigests, respectively.

3.1.6.1.2 Queue

3.1.6.1.2.1 Preconditions

The arguments supplied to the [Create Directory Object \(section 3.1.6.1\)](#) event MUST meet these requirements:

- If *iAttributeList* is provided, the name "Pathname" MUST be present in the list.
- If *iAttributeList* is provided, it MUST NOT contain the name "Scope". This name is not supported in Active Directory-based environments.
- *iDirectoryObject.Pathname* MUST be populated.

If any of these conditions is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.1.2.2 Creation

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **Queue** ADM element attributes listed in section [3.1.6.20.2](#).
2. Let *QueueAttributeList* be a list of mSMQQueue attribute names, initialized to be empty. For each mSMQQueue attribute listed in the following table, if the corresponding **Queue** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the mSMQQueue attribute name to *QueueAttributeList* and compute the value for the attribute as shown. **Queue** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored.

| mSMQQueue attribute | Attribute value computation |
|--|---|
| mSMQLabelEx ([MS-ADA2] section 2.523) | Directory string transformed from <i>iDirectoryObject.Label</i> . |
| mSMQQueueType ([MS-ADA2] section 2.538) | GUID copied from <i>iDirectoryObject.Type</i> . |
| mSMQJournal ([MS-ADA2] section 2.520) | If <i>iDirectoryObject.Journaling</i> is TRUE, mSMQJournal is 1 Else mSMQJournal is 0. |
| mSMQQueueQuota ([MS-ADA2] section 2.537) | Integer number of kilobytes equal to <i>iDirectoryObject.Quota</i> . |
| mSMQQueueJournalQuota ([MS-ADA2] section 2.535) | Integer number of kilobytes equal to <i>iDirectoryObject.JournalQuota</i> . |
| mSMQAuthenticate ([MS-ADA2] section 2.503) | If <i>iDirectoryObject.Authentication</i> is TRUE, mSMQAuthenticate is 1 Else mSMQAuthenticate is 0. |
| mSMQPrivacyLevel ([MS-ADA2] section 2.533) | Computed from <i>iDirectoryObject.PrivacyLevel</i> . See section 3.1.6.1.2.4 . |

| mSMQQueue attribute | Attribute value computation |
|--|---|
| mSMQTransactional ([MS-ADA2] section 2.556) | If <i>iDirectoryObject.Transaction</i> is TRUE, mSMQTransactional is 1 Else mSMQTransactional is 0. |
| MSMQ-MulticastAddress ([MS-ADA2] section 2.500) | Directory string transformed from <i>iDirectoryObject.MulticastAddress</i> . |
| nTSecurityDescriptor ([MS-ADA3] section 2.37) | SECURITY_DESCRIPTOR ([MS-DTYP] section 2.4.6) structure copied from <i>iDirectoryObject.Security</i> . |
| mSMQBasePriority ([MS-ADA2] section 2.504) | Integer copied from <i>iDirectoryObject.BasePriority</i> . |

3. The queue name MUST be modified as specified in section [3.1.6.1.2.5](#).
4. A [Create Object Using LDAP \(section 3.1.6.12\)](#) event MUST be generated with the following arguments:
 - *iParentPath* := a distinguished name of the form specified for an mSMQConfiguration object in section [2.2.1](#), where "<computer name>" is the computer name portion of the value of *iDirectoryObject.Pathname*, as specified in [\[MS-MQMQ\]](#) section 2.1.1.
 - *iChildName* := *FinalChildName*, as computed in step 3.
 - *iObjectClass* := "mSMQQueue"
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *QueueAttributeList* and the corresponding values, as computed in step 2.
5. If the Create Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Create Object Using LDAP event, *rObjectGUID* is undefined, and processing MUST end. Otherwise, *rObjectGuid* is set to the *rObjectGuid* returned by the Create Object Using LDAP event.
6. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.1.2.3 Postprocessing

None.

3.1.6.1.2.4 mSMQPrivacyLevel

The mSMQPrivacyLevel attribute is an integer that MUST have one of the values listed in the following table. The **PrivacyLevel** ADM attribute of the **Queue** ADM element is an enumeration. The value MUST be converted according to this table.

| mSMQPrivacyLevel value | PrivacyLevel ADM attribute value |
|------------------------|----------------------------------|
| 0 | None |
| 1 | Optional |
| 2 | Body |

3.1.6.1.2.5 Name

The name of the queue is modified before the mSMQQueue object is created. The modified name is used to create the object. The modification is performed according to the following algorithm:

1. Let *FinalChildName* and *FinalExtension* be Unicode strings, initialized to be empty.
2. Let *QName* be a Unicode string that MUST be the "<queue name>" portion of *iDirectoryObject.Pathname*, as specified in section [2.2.1](#).
3. A Unicode backslash ('\') MUST be inserted in *QName* before every instance of a character belonging to this set of special characters: '\n', '/', '#', '>', '<', '=', and '\x0A'.
4. If the length of *QName* does not exceed 63 characters, the value of *QName* MUST be copied to *FinalChildName*. Otherwise, the first 55 characters of *QName* MUST be copied to *FinalChildName*; the remaining characters of *QName* MUST be copied to *FinalExtension*; and a nine-character extension MUST be added to the end of *FinalChildName*, which MUST be exactly 64 characters long after adding the extension. The extension consists of a Unicode dash ('-') followed by an eight-character hash string that is calculated from the values of the Unicode characters in *QName* as specified in section [2.2.5](#).
5. If *FinalExtension* is not empty, the mSMQQueueNameExt attribute ([\[MS-ADA2\]](#) section 2.536) MUST be added to *QueueAttributeList*, with the value of *FinalExtension* transformed to a directory string as the associated value.

3.1.6.1.3 Site

3.1.6.1.3.1 Preconditions

The arguments supplied to the [Create Directory Object \(section 3.1.6.1\)](#) event MUST meet these requirements:

- If *iAttributeList* is provided, the name "Name" MUST be present in the list.
- The *iDirectoryObject.Name* ADM attribute MUST be populated.
- *iAttributeList* MUST NOT contain the names **PrimarySiteController** or **PublicSigningKey**. These two ADM element attributes are not supported in Active Directory-based environments.

If any of these conditions is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.1.3.2 Creation

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **Site** ADM element attributes listed in section [3.1.6.20.4](#), except **PrimarySiteController** and **PublicSigningKey**.
2. Let *SiteAttributeList* be a list of site attribute names, initialized to be empty. For each site attribute listed in the following table, if the corresponding **Site** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the site attribute name to *SiteAttributeList* and compute the value of the attribute as shown. **Site** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored.

| site attribute | Attribute value computation |
|---|---|
| mSMQInterval1 ([MS-ADA2] section 2.518) | Integer copied from <i>iDirectoryObject</i> . IntraSiteReplicationInterval . |
| mSMQInterval2 ([MS-ADA2] section 2.519) | Integer copied from <i>iDirectoryObject</i> . InterSiteReplicationInterval . |
| mSMQSiteForeign ([MS-ADA2] section 2.549) | If <i>iDirectoryObject</i> . ForeignSite is TRUE, mSMQSiteForeign is 0x01 Else mSMQSiteForeign is 0x00. |
| mSMQnt4Stub ([MS-ADA2] section 2.528) | If <i>iDirectoryObject</i> . MigratedFromMsmq10 is TRUE, mSMQnt4Stub is 0x01 Else mSMQnt4Stub is 0x00. |
| nTSecurityDescriptor ([MS-ADA3] section 2.37) | See section 3.1.6.1.3.4 . |

3. A [Create Object Using LDAP \(section 3.1.6.12\)](#) event MUST be generated with the following arguments:
 - *iParentPath* := "CN=Services, CN=Configuration, <root>", where <root> is as specified in section [2.2.1](#).
 - *iChildName* := *iDirectoryObject*.**Name**
 - *iObjectClass* := "site"
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *SiteAttributeList* and the corresponding values, as computed in step 2.
4. If the Create Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Create Object Using LDAP event, *rObjectGUID* is undefined, and processing MUST end. Otherwise, *rObjectGuid* is set to the *rObjectGuid* returned by the Create Object Using LDAP event.
5. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.1.3.3 Postprocessing

None.

3.1.6.1.3.4 nTSecurityDescriptor

If *iAttributeList* (either supplied or constructed) includes the ADM attribute name **Security**, the value of the *iDirectoryObject*.**Security** ADM attribute MUST be modified before it is written to the directory. If *iAttributeList* does not include the ADM attribute name **Security**, a default value MUST be computed and written. The algorithm for doing both is as follows:

1. Let *FinalSecurity* and *SuppliedSecurity* be **SECURITY_DESCRIPTOR** structures, as specified in [\[MS-DTYP\]](#) section 2.4.6, initialized to be empty.

2. If *iAttributeList* includes the ADM attribute name **Security**, the value of *iDirectoryObject.Security* MUST be copied to the *SuppliedSecurity* structure.
3. Let *OwnerSid* be a **SID** ([\[MS-DTYP\]](#) section 2.4.2) structure initialized to zero.
4. If the *SuppliedSecurity* structure is not empty:
 1. *OwnerSid* MUST be set to the owner SID from the *SuppliedSecurity* structure.
 2. If the **Dacl** field is populated in the *SuppliedSecurity* structure, the value MUST be copied to the *FinalSecurity* structure.
 3. If the **Sacl** field is populated in the *SuppliedSecurity* structure, the value MUST be copied to the *FinalSecurity* structure.
5. If *OwnerSid* is zero, it MUST be set to the SID of the user under whose identity the current thread is running.
6. If the user referenced by the SID in the *OwnerSid* structure is not a domain user, the *OwnerSid* structure MUST be set to the well-known SID with string representation S-1-5-7 (relative identifier SECURITY_ANONYMOUS_LOGON_RID combined with identifier authority SECURITY_NT_AUTHORITY).
7. If the **Dacl** field was not copied to the *FinalSecurity* structure in step 4:
 1. Let *WorldAccess* and *OwnerAccess* be **MQSITEACCESSMASK** ([\[MS-MQMQ\]](#) section 2.2.23) enumerated values, initialized to zero.
 2. If *OwnerSid* is a guest SID (equal to the SID designated by [DOMAIN_USER_RID_GUEST](#)) or the user referenced by the SID in *OwnerSid* is not a domain user, *WorldAccess* MUST be set to MQSEC_SITE_GENERIC_ALL. Otherwise, *WorldAccess* MUST be set to MQSEC_SITE_GENERIC_READ, and *OwnerAccess* MUST be set to MQSEC_SITE_GENERIC_ALL.
 3. An **ACCESS_ALLOWED_ACE** ([\[MS-DTYP\]](#) section 2.4.4.2) structure with a **Mask** field set to *WorldAccess* and containing the well-known SID with string representation S-1-1-0 (relative identifier SECURITY_WORLD_RID combined with identifier authority SECURITY_WORLD_SID_AUTHORITY) MUST be added to the **Dacl** field in the *FinalSecurity* structure.
 4. If *OwnerAccess* is nonzero, an **ACCESS_ALLOWED_ACE** structure with a **Mask** field set to *OwnerAccess* and containing the SID in *OwnerSid* MUST be added to the **Dacl** field in the *FinalSecurity* structure.
8. The **SECURITY_DESCRIPTOR** structure in *FinalSecurity* MUST be converted to self-relative format (see [\[MS-DTYP\]](#) section 2.4.6).
9. The value of the nTSecurityDescriptor attribute MUST be the value of the *FinalSecurity* structure.

3.1.6.1.4 RoutingLink

3.1.6.1.4.1 Preconditions

The arguments supplied to the [Create Directory Object \(section 3.1.6.1\)](#) event MUST meet these requirements:

- If *iAttributeList* is provided, the names "Site1Identifier", "Site2Identifier", and "ActualCost" MUST be present in the list.

- The *iDirectoryObject.Site1Identifier*, *iDirectoryObject.Site2Identifier*, and *iDirectoryObject.ActualCost* ADM attributes MUST be populated.

If any of these conditions is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.1.4.2 Creation

1. If *iAttributeList* was not provided as an argument, then an *iAttributeList* MUST be constructed that MUST contain the names of all **RoutingLink** ADM element attributes listed in section [3.1.6.20.5](#).
2. Let *SiteLinkAttributeList* be a list of mSMQSiteLink attribute names, initialized to be empty. For each mSMQSiteLink attribute listed in the following table, if the corresponding **RoutingLink** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the mSMQSiteLink attribute name to *SiteLinkAttributeList* and compute the value for the attribute as shown. **RoutingLink** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored.

| mSMQSiteLink attribute | Attribute value computation |
|--|--|
| description ([MS-ADA1] section 2.153) | Directory string transformed from <i>iDirectoryObject.Description</i> |
| mSMQCost ([MS-ADA2] section 2.507) | Integer copied from <i>iDirectoryObject.ActualCost</i> |
| mSMQSite1 ([MS-ADA2] section 2.547) | Computed from <i>iDirectoryObject.Site1Identifier</i> . See section 3.1.6.1.4.4 . |
| mSMQSite2 ([MS-ADA2] section 2.548) | Computed from <i>iDirectoryObject.Site2Identifier</i> . See section 3.1.6.1.4.5 . |
| mSMQSiteGates ([MS-ADA2] section 2.550) | Computed from <i>iDirectoryObject.SiteGateIdentifierList</i> . See section 3.1.6.1.4.6 . |

3. A [Create Object Using LDAP \(section 3.1.6.12\)](#) event MUST be generated with the following arguments:
 - *iParentPath* := "CN=MsmqServices, CN=Services, CN=Configuration, <root>", where <root> is as specified in section [2.2.1](#).
 - *iChildName* := a string composed from the values of *iDirectoryObject.Site1Identifier* and *iDirectoryObject.Site2Identifier*. The two GUIDs MUST be converted to curly braced GUID string form ([\[MS-DTYP\]](#) section 2.3.4.3); next, they MUST be compacted by removing all characters that are not hexadecimal digits; finally, the string for Site2 MUST be appended to the string for Site1.
 - *iObjectClass* := "mSMQSiteLink"
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *SiteLinkAttributeList* and the corresponding values, as computed in step 2.
4. If the Create Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Create

Object Using LDAP event, *rObjectGUID* is undefined, and processing MUST end. Otherwise, *rObjectGuid* is set to the *rObjectGuid* returned by the Create Object Using LDAP event.

5. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.1.4.3 Postprocessing

None.

3.1.6.1.4.4 mSMQSite1

The value of the mSMQSite1 attribute MUST be computed according to the following algorithm:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS *iDirectoryObject.Site1Identifier*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **FullPath**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing MUST end.
3. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of mSMQSite1 MUST be the value of the *rDirectoryObject.FullPath* returned by the Read Directory event.

3.1.6.1.4.5 mSMQSite2

The value of the mSMQSite2 attribute MUST be computed according to the following algorithm:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS *iDirectoryObject.Site2Identifier*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **FullPath**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing MUST end.

3. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of *mSMQSite2* MUST be the value of the *rDirectoryObject.FullPath* returned by the Read Directory event.

3.1.6.1.4.6 mSMQSiteGates

The value of the *mSMQSiteGates* attribute MUST be computed according to the following algorithm:

1. Let *DNList* be a list of distinguished names, initialized to be empty.
2. For each GUID in *iDirectoryObject.SiteGateIdentifierList*, these steps MUST be performed:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from *iDirectoryObject.SiteGateIdentifierList*
 - *iAttributeList* := An array of the following **QueueManager** ADM element attribute names:
 - **FullPath**
 - If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, this GUID MUST be skipped.
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of the *rDirectoryObject.FullPath* returned by the Read Directory event MUST be added to *DNList*.
3. The value of *mSMQSiteGates* MUST be the value of *DNList*.

3.1.6.2 Delete Directory Object

This event MUST be generated with the following arguments:

- *iDirectoryObject*: A **DirectoryObject** ADM element instance to be deleted from the directory.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.

The algorithm MUST perform the following actions to process this event:

- Determine the type of *iDirectoryObject*. If the type is not one of **QueueManager**, **Queue**, **Site**, or **RoutingLink**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
- Perform the steps of the delete operation for that type of ADM element, as specified in the sections listed following, and return the *rStatus* value specified in that section.
 - [QueueManager \(section 3.1.6.2.1\)](#)
 - [Queue \(section 3.1.6.2.2\)](#)
 - [Site \(section 3.1.6.2.3\)](#)

- [RoutingLink \(section 3.1.6.2.4\)](#)

3.1.6.2.1 QueueManager

3.1.6.2.1.1 Preconditions

The arguments supplied to the [Delete Directory Object \(section 3.1.6.2\)](#) event MUST meet the following condition:

- At least one of *iDirectoryObject*.**FullPath**, *iDirectoryObject*.**ComputerName**, or *iDirectoryObject*.**Identifier** MUST be populated.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing of the event MUST end.

3.1.6.2.1.2 Delete

1. Let *DeleteDN* be a distinguished name, initialized to be empty.
2. If *iDirectoryObject*.**FullPath** is populated, the value of *iDirectoryObject*.**FullPath** MUST be copied to *DeleteDN*.
3. If *DeleteDN* is empty and *iDirectoryObject*.**ComputerName** is populated, a distinguished name for the mSMQConfiguration object to be deleted MUST be constructed according to the format specified in section [2.2.1](#), where "<computer name>" is the value of *iDirectoryObject*.**ComputerName**.
4. A [Delete Object Using LDAP \(section 3.1.6.13\)](#) event MUST be generated with the following arguments:
 - *iPath* := *DeleteDN*
 - *iGuid* := *iDirectoryObject*.**Identifier**
5. If the Delete Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Delete Object Using LDAP event, and processing MUST end.

3.1.6.2.1.3 Postprocessing

1. If one or more of *iDirectoryObject*.**RoutingServer**, *iDirectoryObject*.**DirectoryServer**, or *iDirectoryObject*.**SupportingServer** is TRUE, for each GUID in *iDirectoryObject*.**SiteIdentifierList**, these steps MUST be performed:
 1. Search the Directory for a **Site** ADM element instance that has an **Identifier** ADM attribute that matches the GUID:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from the **SiteIdentifierList**
 - *iAttributeList* := An array of the following **Site** ADM attribute names:

- **Name**
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.ObjectNotFound**, the current GUID MUST be skipped.
 - If *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
2. A distinguished name MUST be constructed, of the form specified for an mSMQSettings object in section [2.2.1](#), where "<computer name>" is the value of *iDirectoryObject.ComputerName*, and "<site name>" is the value of the **Name** ADM attribute of the **Site** ADM element instance found in the preceding step.
 3. Generate a [Delete Object Using LDAP \(section 3.1.6.13\)](#) event with the following arguments:
 - *iPath* := the distinguished name constructed in the preceding step.
 4. If the Delete Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Delete Object Using LDAP event, and processing MUST end.
2. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.2.2 Queue

3.1.6.2.2.1 Preconditions

The arguments supplied to the [Delete Directory Object \(section 3.1.6.2\)](#) event MUST meet the following condition:

- At least one of *iDirectoryObject.FullPath*, *iDirectoryObject.Pathname*, or *iDirectoryObject.Identifier* MUST be populated.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing of the event MUST end.

3.1.6.2.2.2 Delete

The delete operation is performed in these steps:

1. Let *DeleteDN* be a distinguished name initialized to be empty.
2. If *iDirectoryObject.FullPath* is populated, the value of *iDirectoryObject.FullPath* MUST be copied to *DeleteDN*.
3. If *DeleteDN* is empty and *iDirectoryObject.Pathname* is populated, a distinguished name for the mSMQQueue object to be deleted MUST be constructed according to the format specified in section [2.2.1](#), where "<computer name>" is the ComputerName portion of the value of *iDirectoryObject.Pathname* and "<queue name>" is the QueueName portion of the value of *iDirectoryObject.Pathname*, as specified in [\[MS-MQMQ\]](#) section 2.1.1.
4. A [Delete Object Using LDAP \(section 3.1.6.13\)](#) event MUST be generated with the following arguments:
 - *iPath* := *DeleteDN*

- *iGuid* := *iDirectoryObject*.**Identifier**

5. If the Delete Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Delete Object Using LDAP event, and processing MUST end.
6. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.2.2.3 Postprocessing

None.

3.1.6.2.3 Site

3.1.6.2.3.1 Preconditions

The arguments supplied to the [Delete Directory Object \(section 3.1.6.2\)](#) event MUST meet the following condition:

- At least one of *iDirectoryObject*.**FullPath**, *iDirectoryObject*.**Name**, or *iDirectoryObject*.**Identifier** MUST be populated.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing of the event MUST end.

3.1.6.2.3.2 Delete

1. Let *DeleteDN* be a distinguished name initialized to be empty.
2. If *iDirectoryObject*.**FullPath** is populated, the value of *iDirectoryObject*.**FullPath** MUST be copied to *DeleteDN*.
3. If *DeleteDN* is empty and *iDirectoryObject*.**Name** is populated, a distinguished name for the site object to be deleted MUST be constructed according to the format specified in section [2.2.1](#), where "<site name>" is the value of *iDirectoryObject*.**Name**.
4. A [Delete Object Using LDAP \(section 3.1.6.13\)](#) event MUST be generated with the following arguments:
 - *iPath* := *DeleteDN*
 - *iGuid* := *iDirectoryObject*.**Identifier**
5. If the Delete Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Delete Object Using LDAP event, and processing MUST end.
6. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.2.3.3 Postprocessing

None.

3.1.6.2.4 RoutingLink

3.1.6.2.4.1 Preconditions

The arguments supplied to the [Delete Directory Object \(section 3.1.6.2\)](#) event MUST meet the following condition:

- At least one of *iDirectoryObject.FullPath* or *iDirectoryObject.Identifier* MUST be populated.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing of the event MUST end.

3.1.6.2.4.2 Delete

1. A [Delete Object Using LDAP \(section 3.1.6.13\)](#) event MUST be generated with the following arguments:
 - *iPath* := *iDirectoryObject.FullPath*
 - *iGuid* := *iDirectoryObject.Identifier*
2. If the Delete Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Delete Object Using LDAP event, and processing MUST end.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.2.4.3 Postprocessing

None.

3.1.6.3 Read Directory

This event MUST be generated with the following arguments:

- *iDirectoryObjectType*: A string that specifies the name of the sub-type of **DirectoryObject** ADM element to be read from the directory.
- *iFilter*: An array of **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20) where the valid attributes are the set of directory ADM attributes associated with a **DirectoryObject** ADM element instance of type *iDirectoryObjectType*. Each **DirectoryObject** ADM element instance that is returned by the read operation MUST satisfy all attribute-filter expressions in this array.
- *iAttributeList (Optional)*: An array of strings containing the names of the directory attributes associated with a **DirectoryObject** ADM element instance of type *iDirectoryObjectType* to be read from the directory and included in the *rDirectoryObject* return value. If this argument is not supplied, all directory ADM attributes are read.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.
- *rDirectoryObject*: A **DirectoryObject** ADM element instance of type *iDirectoryObjectType* populated with the attributes specified in *iAttributeList*, the attributes of which match the expression specified by *iFilter*. This value is undefined if *rStatus* does not equal **DirectoryOperationResult.Success**.

The algorithm MUST perform the following operations to process this event:

1. If the *iFilter* specified contains more than one element, or if it contains one element and that element is not of any of the forms listed in the following table, perform the steps specified in section [3.1.6.3.1](#) and return the *rStatus* value specified in that section. The directory contents MUST NOT be modified by this operation.

| iFilter form |
|---|
| "Identifier" EQUALS <i>aGuid</i> |
| "FullPath" EQUALS <i>aDistinguishedName</i> |

2. Let *UseDN* be a string variable initialized to be empty. If the single element in *iFilter* is of the form "FullPath" EQUALS *aDistinguishedName*, set the value of *UseDN* to *aDistinguishedName*.
3. Let *UseGuid* be a GUID variable initialized to be empty. If the single element in *iFilter* is of the form "Identifier" EQUALS *aGuid*, set the value of *UseGuid* to *aGuid*.
4. If *iDirectoryObjectType* has the value **ConnectedNetwork**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.
5. Perform the steps of the read operation for the type of ADM element indicated by *iDirectoryObjectType*, as specified in the following sections, and return the *rStatus* and *rDirectoryObject* values specified in that section. The directory contents must NOT be modified by this operation.
 - [QueueManager \(section 3.1.6.3.2\)](#)
 - [Queue \(section 3.1.6.3.3\)](#)
 - [Enterprise \(section 3.1.6.3.4\)](#)
 - [Site \(section 3.1.6.3.5\)](#)
 - [RoutingLink \(section 3.1.6.3.6\)](#)
 - [User \(section 3.1.6.3.7\)](#)

3.1.6.3.1 Search For One Object

1. Generate a [Read Directory Begin \(section 3.1.6.4\)](#) event with the following arguments:
 - *iDirectoryObjectType* := *iDirectoryObjectType*
 - *iFilter* := *iFilter*
 - *iAttributeList* := *iAttributeList*
 - *iAttributeSortOrder* := Null
2. If the *rStatus* value returned by the Read Directory Begin event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Read Directory Begin event, and processing MUST end.

3. Let *IteratorHandle* be a **HANDLE** ([\[MS-DTYP\]](#) section 2.2.16) that MUST be initialized to the value returned by the Read Directory Begin (section 3.1.6.4) event in *rQueryHandle*.
4. Generate a [Read Directory Next \(section 3.1.6.5\)](#) event with the following arguments:
 - *iQueryHandle* := *IteratorHandle*
5. If the *rStatus* value returned by the Read Directory Next event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Read Directory Next event, and processing MUST end.
6. Let *DirObj* be a **DirectoryObject** ADM element instance that MUST be initialized to the value returned by the Read Directory Next event in *rDirectoryObject*.
7. Generate a [Read Directory End \(section 3.1.6.6\)](#) event with the following arguments:
 - *iQueryHandle* := *IteratorHandle*
8. If the *rStatus* value returned by the Read Directory End event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Read Directory End event, and processing MUST end.
9. *rStatus* MUST be set to **DirectoryOperationResult.Success**, *rDirectoryObject* MUST be set to the value of *DirObj*, and processing MUST end.

3.1.6.3.2 QueueManager

3.1.6.3.2.1 Preconditions

The arguments supplied to the [Read Directory \(section 3.1.6.3\)](#) event MUST meet the following requirement:

- If *iAttributeList* is provided, it MUST NOT contain the name **ConnectedNetworkIdentifierList**. This attribute is not supported in Active Directory-based environments.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.3.2.2 Read

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **QueueManager** ADM element attributes listed in section [3.1.6.20.1](#), except **ConnectedNetworkIdentifierList**.
2. If any of the ADM attribute names **QualifiedComputerName**, **PublicSigningKeyList**, **OperatingSystemVersion**, or **Clustered** appears in *iAttributeList*, the ADM attribute name **FullPath** MUST be added to *iAttributeList* if it is not already present.
3. If the ADM attribute name **DirectoryServerType** appears in *iAttributeList*, the ADM attribute name **DirectoryServer** MUST be added to *iAttributeList* if it is not already present.
4. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iADMAAttributeList* := *iAttributeList*

5. Let *ConfigurationAttributeList* be a list of mSMQConfiguration attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event.
6. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iGuid* := *UseGuid*
 - *iPath* := *UseDN*
 - *iAttributes* := *ConfigurationAttributeList*
7. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rDirectoryObject* is undefined, and processing MUST end.
8. A [Create ADM Element From LDAP Values \(section 3.1.6.11\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iADMAAttributes* := *iAttributeList*
 - *iLDAPAttributeList* := *ConfigurationAttributeList*
 - *iValues* := *rValues* returned by the Get Object Properties Using LDAP event
9. *rStatus* MUST be set to the *rStatus* returned by the Create ADM Element From LDAP Values event. *rDirectoryObject* MUST be set to the *rDirectoryObject* returned by the Create ADM Element From LDAP Values event. Processing MUST end.

3.1.6.3.3 Queue

3.1.6.3.3.1 Preconditions

None.

3.1.6.3.3.2 Read

1. If *iAttributeList* is not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **Queue** ADM element attributes listed in section [3.1.6.20.2](#).
2. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "Queue"
 - *iADMAAttributeList* := *iAttributeList*
3. Let *QueueAttributeList* be a list of mSMQQueue attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event.
4. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iGuid* := *UseGuid*
 - *iPath* := *UseDN*

- *iAttributes* := *QueueAttributeList*
5. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rDirectoryObject* is undefined, and processing MUST end.
 6. A [Create ADM Element From LDAP Values \(section 3.1.6.11\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "Queue"
 - *iADMAAttributes* := *iAttributeList*
 - *iLDAPAttributeList* := *QueueAttributeList*
 - *iValues* := *rValues* returned by the Get Object Properties Using LDAP event
 7. *rStatus* MUST be set to the *rStatus* returned by the Create ADM Element From LDAP Values event. *rDirectoryObject* MUST be set to the *rDirectoryObject* returned by the Create ADM Element From LDAP Values event. Processing MUST end.

3.1.6.3.4 Enterprise

3.1.6.3.4.1 Preconditions

None.

3.1.6.3.4.2 Read

1. If *iAttributeList* is not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **Enterprise** ADM element attributes listed in section [3.1.6.20.3](#).
2. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "Enterprise"
 - *iADMAAttributeList* := *iAttributeList*
3. Let *EnterpriseAttributeList* be a list of mSMQEnterpriseSettings attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event.
4. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iGuid* := *UseGuid*
 - *iPath* := *UseDN*
 - *iAttributes* := *EnterpriseAttributeList*
5. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rDirectoryObject* is undefined, and processing MUST end.
6. A [Create ADM Element From LDAP Values \(section 3.1.6.11\)](#) event MUST be generated with the following arguments:

- *iDirectoryObjectType* := "Enterprise"
 - *iADMAAttributes* := *iAttributeList*
 - *iLDAPAttributeList* := *EnterpriseAttributeList*
 - *iValues* := *rValues* returned by the Get Object Properties Using LDAP event
7. *rStatus* MUST be set to the *rStatus* returned by the Create ADM Element From LDAP Values event. *rDirectoryObject* MUST be set to the *rDirectoryObject* returned by the Create ADM Element From LDAP Values event. Processing MUST end.

3.1.6.3.5 Site

3.1.6.3.5.1 Preconditions

The arguments supplied to the [Read Directory \(section 3.1.6.3\)](#) event MUST meet the following requirement:

- If *iAttributeList* is provided, it MUST NOT contain the names **PrimarySiteController** or **PublicSigningKey**. These **Site** ADM element attributes are not supported in Active Directory-based environments.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.3.5.2 Read

1. If *iAttributeList* is not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **Site** ADM element attributes listed in section [3.1.6.20.4](#), except **PrimarySiteController** and **PublicSigningKey**.
2. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iADMAAttributeList* := *iAttributeList*
3. Let *SiteAttributeList* be a list of site attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event.
4. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iGuid* := *UseGuid*
 - *iPath* := *UseDN*
 - *iAttributes* := *SiteAttributeList*
5. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rDirectoryObject* is undefined, and processing MUST end.
6. A [Create ADM Element From LDAP Values \(section 3.1.6.11\)](#) event MUST be generated with the following arguments:

- *iDirectoryObjectType* := "Site"
 - *iADMAAttributes* := *iAttributeList*
 - *iLDAPAttributeList* := *SiteAttributeList*
 - *iValues* := *rValues* returned by the Get Object Properties Using LDAP event
7. *rStatus* MUST be set to the *rStatus* returned by the Create ADM Element From LDAP Values event. *rDirectoryObject* MUST be set to the *rDirectoryObject* returned by the Create ADM Element From LDAP Values event. Processing MUST end.

3.1.6.3.6 RoutingLink

3.1.6.3.6.1 Preconditions

None.

3.1.6.3.6.2 Read

1. If *iAttributeList* is not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **RoutingLink** ADM element attributes listed in section [3.1.6.20.5](#).
2. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "RoutingLink"
 - *iADMAAttributeList* := *iAttributeList*
3. Let *SiteLinkAttributeList* be a list of mSMQSiteLink attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event.
4. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iGuid* := *UseGuid*
 - *iPath* := *UseDN*
 - *iAttributes* := *SiteLinkAttributeList*
5. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rDirectoryObject* is undefined, and processing MUST end.
6. A [Create ADM Element From LDAP Values \(section 3.1.6.11\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "RoutingLink"
 - *iADMAAttributes* := *iAttributeList*
 - *iLDAPAttributeList* := *SiteLinkAttributeList*
 - *iValues* := *rValues* returned by the Get Object Properties Using LDAP event

7. *rStatus* MUST be set to the *rStatus* returned by the Create ADM Element From LDAP Values event. *rDirectoryObject* MUST be set to the *rDirectoryObject* returned by the Create ADM Element From LDAP Values event. Processing MUST end.

3.1.6.3.7 User

3.1.6.3.7.1 Preconditions

None.

3.1.6.3.7.2 Read

1. If *iAttributeList* is not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **User** ADM element attributes listed in section [3.1.6.20.6](#).
2. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "User"
 - *iADMAAttributeList* := *iAttributeList*
3. Let *UserAttributeList* be a list of user attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event.
4. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iGuid* := *UseGuid*
 - *iPath* := *UseDN*
 - *iAttributes* := *UserAttributeList*
5. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rDirectoryObject* is undefined, and processing MUST end.
6. A [Create ADM Element From LDAP Values \(section 3.1.6.11\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "User"
 - *iADMAAttributes* := *iAttributeList*
 - *iLDAPAttributeList* := *UserAttributeList*
 - *iValues* := *rValues* returned by the Get Object Properties Using LDAP event
7. *rStatus* MUST be set to the *rStatus* returned by the Create ADM Element From LDAP Values event. *rDirectoryObject* MUST be set to the *rDirectoryObject* returned by the Create ADM Element From LDAP Values event. Processing MUST end.

3.1.6.4 Read Directory Begin

This event MUST be generated with the following arguments:

- *iDirectoryObjectType*: A string that specifies the name of the sub-type of **DirectoryObject** ADM element to be read from the directory.
- *iFilter*: An array of **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20), where the valid attributes comprise the set of directory attributes associated with a **DirectoryObject** ADM element instance of type *iDirectoryObjectType*. Each object that is returned by the read operation MUST satisfy all attribute-filter expressions in this array.
- *iAttributeList* (Optional): An array of strings containing the names of the directory attributes associated with a **DirectoryObject** ADM element instance of type *iDirectoryObjectType* to be read from the directory and included in the *rDirectoryObject* return value. If this argument is not supplied, all directory attributes are read.
- *iAttributeSortOrder* (Optional): An array of **attribute-sort-order values** (as specified following). The elements of this array have a one-to-one correspondence with the elements of *iAttributeList* array. This array represents the precedence that the corresponding attributes in the *iAttributeList* take when ordering the returned results.

An **attribute-sort-order** value consists of the following:

- **SortPriority**: A numeric value that indicates the order in which results would be sorted with respect to the corresponding attribute. Attributes that have a higher sort priority are given higher precedence when sorting the result set. A value of 0 indicates that the corresponding attribute should not be used when sorting the results.
- **SortAscending**: A Boolean value that indicates whether the corresponding attribute should be sorted in ascending order.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.
- *rQueryHandle*: A **HANDLE** ([\[MS-DTYP\]](#) section 2.2.16) that the invoker can use to iterate through the result set. This value is undefined if *rStatus* does not equal **DirectoryOperationResult.Success**.

The algorithm MUST perform the following actions to process this event:

- If *iDirectoryObjectType* has the value "ConnectedNetwork", *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rQueryHandle* is undefined, and processing MUST end.
- Let *ReadIterator* be a [ReadDirectoryIteratorState](#) (section 3.1.1.2) ADM element instance, which MUST be created. The **ReadDirectoryIteratorState** ADM element instance referenced by *ReadIterator* MUST be added to the [ReadDirectoryIteratorStateCollection](#) (section 3.1.1.3) ADM element. *ReadIterator.Handle* MUST be set to a value that is unique within the **ReadDirectoryIteratorStateCollection** ADM element. The value of *iDirectoryObjectType* MUST be copied to *ReadIterator.DataElementType*.
- Based on *iDirectoryObjectType*, perform the steps of the read-begin operation for that type of ADM element, as specified in the following sections, and return the *rStatus* value specified in that section. The directory contents must NOT be modified by this operation.
 - [QueueManager](#) (section 3.1.6.4.1)
 - [Queue](#) (section 3.1.6.4.2)

- [Enterprise \(section 3.1.6.4.3\)](#)
- [Site \(section 3.1.6.4.4\)](#)
- [RoutingLink \(section 3.1.6.4.5\)](#)
- [User \(section 3.1.6.4.6\)](#)
- If the value of *rStatus* is not **DirectoryOperationResult.Success**, the **ReadDirectoryIteratorState** ADM element instance referenced by *ReadIterator* MUST be removed from the **ReadDirectoryIteratorStateCollection** ADM element and discarded, *rQueryHandle* is undefined, and processing MUST end.
- If the value of *rStatus* is **DirectoryOperationResult.Success**, *rQueryHandle* MUST be set to the value of *ReadIterator.Handle*, and processing MUST end.

3.1.6.4.1 QueueManager

3.1.6.4.1.1 Preconditions

The arguments supplied to the [Read Directory Begin \(section 3.1.6.4\)](#) event MUST meet the following requirement:

- If *iAttributeList* is provided, it MUST NOT contain the name **ConnectedNetworkIdentifierList**. This attribute is not supported in Active Directory-based environments.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.4.1.2 Read Begin

1. If *iAttributeList* is not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **QueueManager** ADM element attributes listed in section [3.1.6.20.1](#), except **ConnectedNetworkIdentifierList**.
2. If any of the ADM attribute names **QualifiedComputerName**, **PublicSigningKeyList**, **OperatingSystemVersion**, or **Clustered** appears in *iAttributeList*, the ADM attribute name **FullPath** MUST be added to *iAttributeList* if it is not already present.
3. If the ADM attribute name **DirectoryServerType** appears in *iAttributeList*, the attribute name **DirectoryServer** MUST be added to *iAttributeList* if it is not already present.
4. The value of *iAttributeList* MUST be copied to *ReadIterator.AttributeList*.
5. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iADMAAttributeList* := *iAttributeList*
6. Let *ConfigurationAttributeList* be a list of mSMQConfiguration attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event. The populated *ConfigurationAttributeList* MUST be copied to *ReadIterator.LDAPAttributeList*. If any ADM attribute present in *iAttributeList* does not appear in the *rLDAPAttributesList* returned by the Create LDAP Attribute List event, the entry in *iAttributeSortOrder* corresponding to that *iAttributeList* entry, if any, MUST be removed.

7. Let *ConfigurationFilterList* be a list of attribute-filter expressions modified to refer to mSMQConfiguration attributes and values, initialized to be empty. For each expression in *iFilter*, add an expression to *ConfigurationFilterList* that consists of:

- The mSMQConfiguration attribute that corresponds to the **QueueManager** ADM element attribute in the original expression, as shown in the following table.
- The same operator as the original expression.
- A value transformed from the value in the original expression as shown in the following table.

If the **QueueManager** ADM element attribute in the original expression does not appear in the following table, that expression MUST be ignored.

| QueueManager ADM element attribute | mSMQConfiguration attribute | Value transformation |
|------------------------------------|---|---|
| Identifier | objectGUID ([MS-ADA3] section 2.44) | Copy |
| QueueManagerVersion | mSMQComputerTypeEx ([MS-ADA2] section 2.506) | Copy |
| OperatingSystemType | mSMQOSType ([MS-ADA2] section 2.529) | See the table in section 3.1.6.1.1.4 . |
| CreateTime | whenCreated ([MS-ADA3] section 2.371) | Converts to midnight (00:00:00), January 1, 1970 UTC + CreateTime seconds. |
| ModifyTime | whenChanged ([MS-ADA3] section 2.370) | Converts to midnight (00:00:00), January 1, 1970 UTC + ModifyTime seconds. |
| QueueManagerQuota | mSMQQuota ([MS-ADA2] section 2.539) | Copy |
| JournalQuota | mSMQJournalQuota ([MS-ADA2] section 2.521) | Copy |
| ForeignSystem | mSMQForeign ([MS-ADA2] section 2.516) | TRUE converts to 0x01; FALSE converts to 0x00. |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) | Copy |
| SiteIdentifierList | mSMQSites ([MS-ADA2] section 2.555) | Copy |
| RoutingServer | mSMQRoutingServices ([MS-ADA2] section 2.541) | TRUE converts to 0x01; FALSE converts to 0x00. |
| DirectoryServer | mSMQDsServices ([MS-ADA2] section 2.514) | TRUE converts to 0x01; FALSE converts to 0x00. |
| SupportingServer | mSMQDependentClientServices ([MS-ADA2] section 2.510) | TRUE converts to 0x01; FALSE converts to 0x00. |

| QueueManager ADM element attribute | mSMQConfiguration attribute | Value transformation |
|------------------------------------|--|----------------------|
| PublicEncryptionKeyList | mSMQEncryptKey ([MS-ADA2] section 2.515) | Copy |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) | Copy |

8. A [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be generated with the following arguments:

- *iObjectClass* := mSMQConfiguration
- *iFilter* := ConfigurationFilterList
- *iAttributes* := ConfigurationAttributeList

9. If the Search Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success** and *rStatus* is set to **DirectoryOperationResult.ObjectNotFound**, *ReadIterator.LDAPState* MUST be set to an empty list, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

10. If the search fails for any other reason, *rStatus* MUST be set to the *rStatus* returned by the Search Using LDAP event, and processing MUST end.

3.1.6.4.1.3 Postprocessing

1. If *iAttributeSortOrder* is supplied, the sublists in the *rValues* list returned by the [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be sorted according to the priorities and orders specified in *iAttributeSortOrder*, as specified at the start of section [3.1.6.4](#), where the priorities and orders apply to the corresponding attributes named in *QueueAttributeList*.
2. The sorted *rValues* list MUST be stored in *ReadIterator.LDAPState*.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.4.2 Queue

3.1.6.4.2.1 Preconditions

None.

3.1.6.4.2.2 Read Begin

1. If *iAttributeList* was not provided as an argument, then an *iAttributeList* MUST be constructed that MUST contain the names of all **Queue** ADM element attributes listed in section [3.1.6.20.2](#).
2. The value of *iAttributeList* MUST be copied to *ReadIterator.AttributeList*.
3. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "Queue"
 - *iADMAtributeList* := *iAttributeList*

4. Let *QueueAttributeList* be a list of mSMQQueue attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event. The populated *QueueAttributeList* MUST be copied to *ReadIterator.LDAPAttributeList*. If any ADM attribute present in *iAttributeList* does not appear in the *rLDAPAttributesList* returned by the Create LDAP Attribute List event, the entry in *iAttributeSortOrder* corresponding to that *iAttributeList* entry, if any, MUST be removed.
5. Let *QueueFilterList* be a list of attribute-filter expressions modified to refer to mSMQQueue attributes and values, initialized to be empty. For each expression in *iFilter*, add an expression to *QueueFilterList* that consists of:
 - The mSMQQueue attribute that corresponds to the **Queue** ADM element attribute in the original expression, as shown in the following table.
 - The same operator as the original expression.
 - A value transformed from the value in the original expression as shown in the following table.

If the **Queue** ADM element attribute in the original expression does not appear in the following table, the expression MUST be ignored.

| Queue ADM element attribute | mSMQQueue attribute | Value transformation |
|-----------------------------|---|---|
| Identifier | objectGUID ([MS-ADA3] section 2.44) | Copy |
| Label | mSMQLabelEx ([MS-ADA2] section 2.523) | Copy |
| CreateTime | whenCreated ([MS-ADA3] section 2.371) | Converts to midnight (00:00:00), January 1, 1970 UTC + CreateTime seconds. |
| ModifyTime | whenChanged ([MS-ADA3] section 2.370) | Converts to midnight (00:00:00), January 1, 1970 UTC + ModifyTime seconds. |
| Type | mSMQQueueType ([MS-ADA2] section 2.538) | Copy |
| Journaling | mSMQJournal ([MS-ADA2] section 2.520) | TRUE converts to 1; FALSE converts to 0. |
| Quota | mSMQQueueQuota ([MS-ADA2] section 2.537) | Copy |
| JournalQuota | mSMQQueueJournalQuota ([MS-ADA2] section 2.521) | Copy |
| Authentication | mSMQAuthenticate ([MS-ADA2] section 2.503) | TRUE converts to 1; FALSE converts to 0. |
| PrivacyLevel | mSMQPrivacyLevel ([MS-ADA2] section 2.533) | See the table in section 3.1.6.1.2.4 . |
| Transactional | mSMQTransactional ([MS-ADA2] section 2.556) | TRUE converts to 1; FALSE converts to 0. |

| Queue ADM element attribute | mSMQQueue attribute | Value transformation |
|-----------------------------|---|----------------------|
| MulticastAddress | MSMQ-MulticastAddress ([MS-ADA2] section 2.500) | Copy |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) | Copy |
| BasePriority | mSMQBasePriority ([MS-ADA2] section 2.504) | Copy |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) | Copy |

6. A [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be generated with the following arguments:
 - *iObjectClass* := mSMQQueue
 - *iFilter* := *QueueFilterList*
 - *iAttributes* := *QueueAttributeList*
7. If the Search Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success** and *rStatus* is set to **DirectoryOperationResult.ObjectNotFound**, *ReadIterator.LDAPState* MUST be set to an empty list, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.
8. If the search fails for any other reason, *rStatus* MUST be set to the *rStatus* returned by the Search Using LDAP event, and processing MUST end.

3.1.6.4.2.3 Postprocessing

1. If *iAttributeSortOrder* is supplied, the sublists in the *rValues* list returned by the [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be sorted according to the priorities and orders specified in *iAttributeSortOrder*, as specified at the start of section 3.1.6.4, where the priorities and orders apply to the corresponding attributes named in *QueueAttributeList*.
2. The sorted *rValues* list MUST be stored in *ReadIterator.LDAPState*.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.4.3 Enterprise

3.1.6.4.3.1 Preconditions

None.

3.1.6.4.3.2 Read Begin

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **Enterprise** ADM element attributes listed in section [3.1.6.20.3](#).
2. The value of *iAttributeList* MUST be copied to *ReadIterator.AttributeList*.

3. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "Enterprise"
 - *iADMAtributeList* := *iAttributeList*
4. Let *EnterpriseAttributeList* be a list of mSMQEnterpriseSettings attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event. The populated *EnterpriseAttributeList* MUST be copied to *ReadIterator.LDAPAttributeList*. If any ADM attribute present in *iAttributeList* does not appear in the *rLDAPAttributesList* returned by the Create LDAP Attribute List event, the entry in *iAttributeSortOrder* corresponding to that *iAttributeList* entry, if any, MUST be removed.
5. Let *EnterpriseFilterList* be a list of attribute-filter expressions modified to refer to mSMQEnterpriseSettings attributes and values, initialized to be empty. For each expression in *iFilter*, add an expression to *EnterpriseFilterList* that consists of:
 - The mSMQEnterpriseSettings attribute that corresponds to the **Enterprise** ADM element attribute in the original expression, as shown in the following table.
 - The same operator as the original expression.
 - A value transformed from the value in the original expression as shown in the following table.

If the **Enterprise** ADM element attribute in the original expression does not appear in the following table, the expression MUST be ignored.

| Enterprise ADM element attribute | mSMQEnterpriseSettings attribute | Value transformation |
|---|--|--|
| Identifier | objectGUID ([MS-ADA3] section 2.44) | Copy |
| WeakenedSecurity | mSMQNameStyle ([MS-ADA2] section 2.526) | See section 3.1.6.4.3.4 . |
| NonLDAPCapableQueueManagerNotification | mSMQCSPName ([MS-ADA2] section 2.508) | See section 3.1.6.4.3.5 . |
| DefaultTimeToLive | mSMQLongLived ([MS-ADA2] section 2.524) | Copy |
| OldDirectory | mSMQVersion ([MS-ADA2] section 2.558) | TRUE converts to 3; FALSE converts to 200. |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) | Copy |

6. A [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be generated with the following arguments:
 - *iObjectClass* := mSMQEnterpriseSettings
 - *iFilter* := *EnterpriseFilterList*
 - *iAttributes* := *EnterpriseAttributeList*

7. If the Search Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success** and *rStatus* is set to **DirectoryOperationResult.ObjectNotFound**, *ReadIterator.LDAPState* MUST be set to an empty list, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.
8. If the search fails for any other reason, *rStatus* MUST be set to the *rStatus* returned by the Search Using LDAP event, and processing MUST end.

3.1.6.4.3.3 Postprocessing

1. If *iAttributeSortOrder* is supplied, the sublists in the *rValues* list returned by the [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be sorted according to the priorities and orders specified in *iAttributeSortOrder*, as specified at the start of section [3.1.6.4](#), where the priorities and orders apply to the corresponding attributes named in *EnterpriseAttributeList*.
2. The sorted *rValues* list MUST be stored in *ReadIterator.LDAPState*.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.4.3.4 WeakenedSecurity

The *mSMQNameStyle* attribute is an integer that MUST have one of the values listed in the following table. The *WeakenedSecurity* attribute is a Boolean. The values MUST be converted according to this table.

| <i>mSMQNameStyle</i> value | <i>WeakenedSecurity</i> value |
|----------------------------|-------------------------------|
| 0x00 | FALSE |
| 0x01 | TRUE |

3.1.6.4.3.5 NonLDAPCapableQueueManagerNotification

The *mSMQCSPName* attribute is a string that MUST have one of the values listed in the following table. The attribute is a Boolean. The values MUST be converted according to this table.

| <i>mSMQCSPName</i> value | <i>NonLDAPCapableQueueManagerNotification</i> ADM attribute value |
|--------------------------|---|
| "Y" | true |
| "N" | false |

3.1.6.4.4 Site

3.1.6.4.4.1 Preconditions

The arguments supplied to the [Read Directory Begin \(section 3.1.6.4\)](#) event MUST meet the following requirement:

- If *iAttributeList* is provided, it MUST NOT contain the names **PrimarySiteController** or **PublicSigningKey**. These **Site** ADM element attributes are not supported in Active Directory-based environments.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.4.4.2 Read Begin

1. If *iAttributeList* was not provided as an argument, then an *iAttributeList* MUST be constructed that MUST contain the names of all **Site** ADM element attributes listed in section [3.1.6.20.4](#), except **PrimarySiteController** and **PublicSigningKey**.
2. The value of *iAttributeList* MUST be copied to *ReadIterator.AttributeList*.
3. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iADMAtributeList* := *iAttributeList*
4. Let *SiteAttributeList* be a list of site attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event. The populated *SiteAttributeList* MUST be copied to *ReadIterator.LDAPAttributeList*. If any ADM attribute present in *iAttributeList* does not appear in the *rLDAPAttributesList* returned by the Create LDAP Attribute List event, the entry in *iAttributeSortOrder* corresponding to that *iAttributeList* entry, if any, MUST be removed.
5. Let *SiteFilterList* be a list of attribute-filter expressions modified to refer to site attributes and values, initialized to be empty. For each expression in *iFilter*, add an expression to *SiteFilterList* that consists of:
 - The site attribute that corresponds to the **Site** ADM element attribute in the original expression, as shown in the following table.
 - The same operator as the original expression.
 - A value transformed from the value in the original expression as shown in the following table.

If the **Site** ADM element attribute in the original expression does not appear in the following table, the expression MUST be ignored.

| Site ADM element attribute | site attribute | Value transformation |
|-------------------------------------|--|---|
| Identifier | objectGUID ([MS-ADA3] section 2.44) | Copy |
| Name | cn ([MS-ADA1] section 2.110) | Copy |
| IntraSiteReplicationInterval | mSMQInterval1 ([MS-ADA2] section 2.518) | Copy |
| InterSiteReplicationInterval | mSMQInterval2 ([MS-ADA2] section 2.519) | Copy |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) | Copy |
| Foreign | mSMQSiteForeign ([MS-ADA2] section 2.549) | TRUE converts to 0x01; FALSE converts to 0x00. |

| Site ADM element attribute | site attribute | Value transformation |
|----------------------------|---|--|
| MigratedFromMsmq10 | mSMQnt4Stub ([MS-ADA2] section 2.528) | TRUE converts to 0x01; FALSE converts to 0x00. |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) | Copy |

6. A [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be generated with the following arguments:
 - *iObjectClass* := site
 - *iFilter* := *SiteFilterList*
 - *iAttributes* := *SiteAttributeList*
7. If the Search Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success** and *rStatus* is set to **DirectoryOperationResult.ObjectNotFound**, *ReadIterator.LDAPState* MUST be set to an empty list, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.
8. If the search fails for any other reason, *rStatus* MUST be set to the *rStatus* returned by the Search Using LDAP event, and processing MUST end.

3.1.6.4.4.3 Postprocessing

1. If *iAttributeSortOrder* is supplied, the sublists in the *rValues* list returned by the [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be sorted according to the priorities and orders specified in *iAttributeSortOrder*, as specified at the start of section 3.1.6.4, where the priorities and orders apply to the corresponding attributes named in *SiteAttributeList*.
2. The sorted *rValues* list MUST be stored in *ReadIterator.LDAPState*.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.4.5 RoutingLink

3.1.6.4.5.1 Preconditions

None.

3.1.6.4.5.2 Read Begin

1. If *iAttributeList* was not provided as an argument, then an *iAttributeList* MUST be constructed that MUST contain the names of all **RoutingLink** ADM element attributes listed in section [3.1.6.20.5](#).
2. The value of *iAttributeList* MUST be copied to *ReadIterator.AttributeList*.
3. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "RoutingLink"
 - *iADMAAttributeList* := *iAttributeList*

4. Let *SiteLinkAttributeList* be a list of mSMQSiteLink attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event. The populated *SiteLinkAttributeList* MUST be copied to *ReadIterator.LDAPAttributeList*. If any ADM attribute present in *iAttributeList* does not appear in the *rLDAPAttributesList* returned by the Create LDAP Attribute List event, the entry in *iAttributeSortOrder* corresponding to that *iAttributeList* entry, if any, MUST be removed.
5. Let *SiteLinkFilterList* be a list of attribute-filter expressions modified to refer to mSMQSiteLink attributes and values, initialized to be empty. For each expression in *iFilter*, add an expression to *SiteLinkFilterList* that consists of:
 - The mSMQSiteLink attribute that corresponds to the **RoutingLink** ADM element attribute in the original expression, as shown in the following table.
 - The same operator as the original expression.
 - A value transformed from the value in the original expression as shown in the following table.

If the **RoutingLink** ADM element attribute in the original expression does not appear in the following table, the expression MUST be ignored.

| RoutingLink ADM element attribute | mSMQSiteLink attribute | Value transformation |
|-----------------------------------|---|---|
| Identifier | objectGUID ([MS-ADA3] section 2.44) | Copy |
| Description | description ([MS-ADA1] section 2.153) | Copy |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) | Copy |
| ActualCost | mSMQCost ([MS-ADA2] section 2.507) | Copy |
| Site1Identifier | mSMQSite1 ([MS-ADA2] section 2.547) | See section 3.1.6.4.5.4 . |
| Site2Identifier | mSMQSite2 ([MS-ADA2] section 2.548) | See section 3.1.6.4.5.5 . |

6. A [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be generated with the following arguments:
 - *iObjectClass* := mSMQSiteLink
 - *iFilter* := *SiteLinkFilterList*
 - *iAttributes* := *SiteLinkAttributeList*
7. If the Search Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success** and *rStatus* is set to **DirectoryOperationResult.ObjectNotFound**, *ReadIterator.LDAPState* MUST be set to an empty list, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.
8. If the search fails for any other reason, *rStatus* MUST be set to the *rStatus* returned by the Search Using LDAP event, and processing MUST end.

3.1.6.4.5.3 Postprocessing

1. If *iAttributeSortOrder* is supplied, the sublists in the *rValues* list returned by the [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be sorted according to the priorities and orders specified in *iAttributeSortOrder*, as specified at the start of section [3.1.6.4](#), where the priorities and orders apply to the corresponding attributes named in *SiteLinkAttributeList*.
2. The sorted *rValues* list MUST be stored in *ReadIterator.LDAPState*.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.4.5.4 Site1Identifier Filtering

The filter value to be used when filtering on the **Site1Identifier** ADM attribute MUST be computed according to the following algorithm:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the value of the original filter value for **Site1Identifier**
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **FullPath**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**:
 - *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
3. The filter value for the **Site1Identifier** ADM attribute MUST be the value of the *rDirectoryObject.FullPath* returned by the Read Directory event.

3.1.6.4.5.5 Site2Identifier Filtering

The filter value to be used when filtering on the **Site2Identifier** ADM attribute MUST be computed according to the following algorithm:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the value of the original filter value for **Site2Identifier**
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **FullPath**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**:

- *rStatus* must be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
3. The filter value for **Site2Identifier** ADM attribute MUST be the value of the *rDirectoryObject.FullPath* returned by the Read Directory event.

3.1.6.4.6 User

3.1.6.4.6.1 Preconditions

None.

3.1.6.4.6.2 Read Begin

1. If *iAttributeList* was not provided as an argument, then an *iAttributeList* MUST be constructed that MUST contain the names of all **User** ADM element attributes listed in section [3.1.6.20.6](#).
2. The value of *iAttributeList* MUST be copied to *ReadIterator.AttributeList*.
3. A [Create LDAP Attribute List \(section 3.1.6.10\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := "User"
 - *iADMAtributeList* := *iAttributeList*
4. Let *UserAttributeList* be a list of user attribute names, initialized to be the LDAP attribute names returned in *rLDAPAttributeList* by the Create LDAP Attribute List event. If any ADM attribute present in *iAttributeList* does not appear in the *rLDAPAttributesList* returned by the Create LDAP Attribute List event, the entry in *iAttributeSortOrder* corresponding to that *iAttributeList* entry, if any, MUST be removed.
5. Let *UserFilterList* be a list of attribute-filter expressions modified to refer to user attributes and values, initialized to be empty. For each expression in *iFilter*, add an expression to *UserFilterList* that consists of:
 - The user attribute that corresponds to the **User** ADM element attribute in the original expression, as shown in the following table.
 - The same operator as the original expression.
 - A value transformed from the value in the original expression as shown in the following table.

If the **User** ADM element attribute in the original expression does not appear in the following table, the expression MUST be ignored.

| User ADM element attribute | user attribute | Value transformation |
|------------------------------|---|----------------------|
| Identifier | objectGUID ([MS-ADA3] section 2.44) | Copy |
| SecurityIdentifier | objectSid ([MS-ADA3] section 2.45) | Copy |
| CertificateDigestList | mSMQDigests ([MS-ADA2] section 2.511) | Copy |
| Certificates | mSMQSignCertificates ([MS-ADA2] section 2.544) | Copy |

| User ADM element attribute | user attribute | Value transformation |
|----------------------------|--|----------------------|
| FullPath | distinguishedName ([MS-ADA1] section 2.177) | Copy |

6. A [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be generated with the following arguments:
 - *iObjectClass* := user
 - *iFilter* := *UserFilterList*
 - *iAttributes* := *UserAttributeList*
7. If the Search Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, and *rStatus* is set to **DirectoryOperationResult.ObjectNotFound**, *ReadIterator.LDAPState* MUST be set to an empty list, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.
8. If the search fails for any other reason, *rStatus* MUST be set to the *rStatus* returned by the Search Using LDAP event, and processing MUST end.

3.1.6.4.6.3 Postprocessing

1. If *iAttributeSortOrder* is supplied, the sublists in the *rValues* list returned by the [Search Using LDAP \(section 3.1.6.15\)](#) event MUST be sorted according to the priorities and orders specified in *iAttributeSortOrder*, as specified at the start of section [3.1.6.4](#), where the priorities and orders apply to the corresponding attributes named in *UserAttributeList*.
2. The sorted *rValues* list MUST be stored in *ReadIterator.LDAPState*.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.5 Read Directory Next

This event MUST be generated with the following arguments:

- *iQueryHandle*: The **HANDLE** that was generated by the [Read Directory Begin \(section 3.1.6.4\)](#) event.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.
- *rDirectoryObject*: The next **DirectoryObject** ADM element instance from the collection of **DirectoryObject** ADM element instances that match the *iFilter* expressions supplied to the Read Directory Begin event that created the supplied *iQueryHandle*. The collection is ordered based on *iAttributeSortOrder* supplied to the Read Directory Begin event that created the supplied *iQueryHandle*. This value is populated with the attributes specified in *iAttributeList* supplied to the Read Directory Begin event that created the supplied *iQueryHandle*. This value is undefined if *rStatus* does not equal **DirectoryOperationResult.Success**.

The algorithm MUST perform the following actions to process this event:

- Let *ReadIterator* be the **ReadDirectoryIteratorState** ADM element instance found in the **ReadDirectoryIteratorStateCollection** ADM element for which the value of the **Handle** ADM

attribute equals the value of *iQueryHandle*. If no such **ReadDirectoryIteratorState** ADM element is found, processing MUST end. *rStatus* MUST be set to **DirectoryOperationResult.GenericError**.

- Let *NextResult* be a list of attribute values initialized by removing the first sublist from *ReadIterator.LDAPState* and placing it in *NextResult*. If *ReadIterator.LDAPState* is empty, *rStatus* MUST be set to **DirectoryOperationResult.EndOfData**, and processing MUST end.
- A [Create ADM Element From LDAP Values \(section 3.1.6.11\)](#) event MUST be generated with the following arguments:
 - *iDirectoryObjectType* := *ReadIterator.DataElementType*
 - *iADMAtributes* := *ReadIterator.AttributeList*
 - *iLDAPAttributeList* := *ReadIterator.LDAPAttributeList*
 - *iValues* := *NextResult*
- *rStatus* MUST be set to the *rStatus* returned by the Create ADM Element From LDAP Values event. *rDirectoryObject* MUST be set to the *rDirectoryObject* returned by the Create ADM Element From LDAP Values event. Processing MUST end.

3.1.6.6 Read Directory End

This event MUST be generated with the following arguments:

- *iQueryHandle*: The **HANDLE** that was generated by the [Read Directory Begin \(section 3.1.6.4\)](#) event.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.

The algorithm MUST perform the following actions to process this event:

- Let *ReadIterator* be the **ReadDirectoryIteratorState** ADM element instance in the **ReadDirectoryIteratorStateCollection** ADM element for which the value of the **Handle** ADM attribute equals the value of *iQueryHandle*. If there is no such **ReadDirectoryIteratorState** ADM element instance in the collection, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
- The **ReadDirectoryIteratorState** ADM element instance referenced by *ReadIterator* MUST be removed from the **ReadDirectoryIteratorStateCollection** ADM element.
- The search results stored in *ReadIterator.LDAPState* MUST be discarded.
- The **ReadDirectoryIteratorState** ADM element instance referenced by *ReadIterator* MUST be discarded.
- *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.7 Write Directory

This event MUST be generated with the following arguments:

- *iDirectoryObject*: The **DirectoryObject** ADM element instance to be written to the directory.

- *iAttributeList* (Optional): An array of strings containing the names of the directory attributes associated with *iDirectoryObject* to be updated to the directory. If this argument is not supplied, all directory attributes are updated.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.

The algorithm MUST perform the following actions to process this event:

- Determine the type of *iDirectoryObject*. If the type is **ConnectedNetwork** ([\[MS-MQDMPR\] section 3.1.1.9](#)), *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
- Perform the steps of the write operation for that type of ADM element, as specified in the sections listed following, and return the *rStatus* value specified in that section.
 - [QueueManager \(section 3.1.6.7.1\)](#)
 - [Queue \(section 3.1.6.7.2\)](#)
 - [Enterprise \(section 3.1.6.7.3\)](#)
 - [Site \(section 3.1.6.7.4\)](#)
 - [RoutingLink \(section 3.1.6.7.5\)](#)
 - [User \(section 3.1.6.7.6\)](#)

3.1.6.7.1 QueueManager

3.1.6.7.1.1 Preconditions

The arguments supplied to the [Write Directory \(section 3.1.6.7\)](#) event MUST meet these requirements:

- If *iAttributeList* is provided, it MUST NOT contain the name **ConnectedNetworkIdentifierList**. This attribute is not supported in Active Directory-based environments.
- At least one of *iDirectoryObject.FullPath*, *iDirectoryObject.ComputerName*, or *iDirectoryObject.Identifier* MUST be populated.

If any of these conditions is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.7.1.2 Write

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **QueueManager** ADM element attributes listed in section [3.1.6.20.1](#), except **ConnectedNetworkIdentifierList**.
2. Let *ConfigurationAttributeList* be a list of mSMQConfiguration attribute names, initialized to be empty. For each mSMQConfiguration attribute listed in the following table, if the corresponding **QueueManager** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the mSMQConfiguration attribute name to *ConfigurationAttributeList* and compute the value for the attribute as shown. **QueueManager**

ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored in this step.

| mSMQConfiguration attribute | Attribute value computation |
|--|--|
| mSMQComputerTypeEx ([MS-ADA2] section 2.506) | Directory string transformed from <i>iDirectoryObject.QueueManagerVersion</i> . |
| mSMQOSType ([MS-ADA2] section 2.529) | Computed from <i>iDirectoryObject.OperatingSystemType</i> . See section 3.1.6.7.1.4 . |
| mSMQServiceType ([MS-ADA2] section 2.543) | Computed from <i>iDirectoryObject.DirectoryServer</i> , <i>iDirectoryObject.DirectoryServerType</i> , <i>iDirectoryObject.RemoteAccessServer</i> , and <i>iDirectoryObject.RoutingServer</i> . See section 3.1.6.7.1.5 . |
| mSMQQuota ([MS-ADA2] section 2.539) | Integer copied from <i>iDirectoryObject.QueueManagerQuota</i> . |
| mSMQJournalQuota ([MS-ADA2] section 2.521) | Integer copied from <i>iDirectoryObject.JournalQuota</i> . |
| mSMQForeign ([MS-ADA2] section 2.516) | If <i>iDirectoryObject.ForeignSystem</i> is TRUE, mSMQForeign is 0x01 Else mSMQForeign is 0x00. |
| mSMQSites ([MS-ADA2] section 2.555) | List of GUIDs copied from <i>iDirectoryObject.SiteIdentifierList</i> . |
| mSMQOutRoutingServers ([MS-ADA2] section 2.530) | Computed from <i>iDirectoryObject.OutRoutingServerIdentifierList</i> . See section 3.1.6.7.1.6 . |
| mSMQInRoutingServers ([MS-ADA2] section 2.517) | Computed from <i>iDirectoryObject.InRoutingServerIdentifierList</i> . See section 3.1.6.7.1.7 . |
| mSMQRoutingServices ([MS-ADA2] section 2.541) | If <i>iDirectoryObject.RoutingServer</i> is TRUE, mSMQRoutingServices is 0x01 Else mSMQRoutingServices is 0x00. |
| mSMQDsServices ([MS-ADA2] section 2.514) | If <i>iDirectoryObject.DirectoryServer</i> is TRUE, mSMQDsServices is 0x01 Else mSMQDsServices is 0x00. |
| mSMQDependentClientServices ([MS-ADA2] section 2.510) | If <i>iDirectoryObject.SupportingServer</i> is TRUE, mSMQDependentClientServices is 0x01 Else mSMQDependentClientServices is 0x00. |
| mSMQEncryptKey ([MS-ADA2] section 2.515) | MQDSPUBLICKEYS ([MS-MOMQ] section 2.2.2) copied from <i>iDirectoryObject.PublicEncryptionKeyList</i> . |

3. Let *WriteGUID* be a distinguished name, initialized to be empty. If *iDirectoryObject.Identifier* is populated, the value of *iDirectoryObject.Identifier* MUST be copied to *WriteGUID*.
4. Let *WriteDN* be a distinguished name, initialized to be empty.
5. If *iDirectoryObject.FullPath* is populated, the value of *iDirectoryObject.FullPath* MUST be copied to *WriteDN*.
6. If *WriteDN* is empty and *iDirectoryObject.ComputerName* is populated, the value of *WriteDN* MUST be a distinguished name for a mSMQConfiguration object, constructed according to the format specified in section 2.2.1, where "<computer name>" is the value of *iDirectoryObject.ComputerName*.
7. A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event MUST be generated with the following arguments:
 - *iPath* := *WriteDN*
 - *iGuid* := *WriteGUID*
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *ConfigurationAttributeList* and the corresponding values, as computed in step 2.
8. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing MUST end.

3.1.6.7.1.3 Postprocessing

1. If *iAttributeList* contains the name "Security" and the *iDirectoryObject.Security* ADM attribute is populated, a [Set Object Security Using LDAP \(section 3.1.6.17\)](#) event MUST be generated with the following arguments:
 - *iPath* := *WriteDN*
 - *iGuid* := *WriteGUID*
 - *iSecurity* := *iDirectoryObject.Security*
2. If the Set Object Security Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Security Using LDAP event, and processing MUST end.
3. If the **PublicSigningKeyList** ADM attribute is present in *iAttributeList*, a [Set Object Properties Using LDAP](#) event MUST be generated with the following arguments:
 - *iPath* := a distinguished name for a computer object of the format specified in section 2.2.1, where "<computer name>" is the value of *iDirectoryObject.ComputerName*.
 - *iAttributes* := a list of attribute name-value pairs, where the names are "mSMQSignCertificates" and "mSMQDigests", and the values are computed as specified in section 3.1.6.7.1.8.
4. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing MUST end.

5. The steps in section [3.1.6.7.1.9](#) MUST be performed to determine whether any mSMQSettings objects are affected and any necessary updates MUST be performed.
6. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.7.1.4 mSMQOSType

The mSMQOSType attribute is an integer that MUST have one of the values listed in the following table. The **OperatingSystemType** ADM attribute is an enumeration. The values MUST be converted according to this table.

| OperatingSystemType ADM attribute value | mSMQOSType value |
|---|------------------|
| Unknown | 0x00000000 |
| Foreign | 0x00000100 |
| Windows 95 | 0x00000200 |
| WinClient | 0x00000300 |
| WinServer | 0x00000400 |
| WinEnt | 0x00000500 |

3.1.6.7.1.5 mSMQServiceType

The mSMQServiceType attribute is a bitmap that MUST be computed according to this algorithm:

1. Let *FinalValue* be a 32-bit unsigned integer, initialized to 0x00000000.
2. If *iDirectoryObject.RoutingServer* is populated and TRUE, the 0x00000001 bit of *FinalValue* MUST be set.
3. If *iDirectoryObject.RemoteAccessServer* is populated and TRUE, the 0x00000010 bit of *FinalValue* MUST be set.
4. If *iDirectoryObject.DirectoryServer* is populated and TRUE, and *iDirectoryObject.DirectoryServerType* is populated and is not set to the enumeration value **Standalone**, one bit of *FinalValue* MUST be set according to the following table.

| DirectoryServerType value | FinalValue bit to be set |
|------------------------------------|--------------------------|
| BackupSiteController | 0x00000002 |
| PrimarySiteController | 0x00000004 |
| PrimaryEnterpriseController | 0x00000008 |

5. The value of the mSMQServiceType attribute MUST be the value of *FinalValue*.

3.1.6.7.1.6 mSMQOutRoutingServers

The value of the mSMQOutRoutingServers attribute MUST be computed according to the following algorithm:

1. Let *DNList* be a list of distinguished names, initialized to be empty.
2. For each GUID in *iDirectoryObject.OutRoutingServerIdentifierList*, these steps MUST be performed:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from *iDirectoryObject.OutRoutingServerIdentifierList*
 - *iAttributeList* := An array of the following **QueueManager** ADM element attribute names:
 - **FullPath**
 - If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, this GUID MUST be skipped.
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of the *rDirectoryObject.FullPath* returned by the Read Directory event must be added to *DNList*.
3. The value of *mSMQOutRoutingServers* MUST be the value of *DNList*.

3.1.6.7.1.7 mSMQInRoutingServers

The value of the *mSMQInRoutingServers* attribute MUST be computed according to the following algorithm:

1. Let *DNList* be a list of distinguished names, initialized to be empty.
2. For each GUID in *iDirectoryObject.InRoutingServerIdentifierList*, these steps MUST be performed:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from *iDirectoryObject.InRoutingServerIdentifierList*
 - *iAttributeList* := An array of the following **QueueManager** ADM element attribute names:
 - **FullPath**
 - If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, this GUID MUST be skipped.
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of the *rDirectoryObject.FullPath* returned by the Read Directory event must be added to *DNList*.

3. The value of `mSMQInRoutingServers` MUST be the value of `DNList`.

3.1.6.7.1.8 mSMQSignCertificates and mSMQDigests

The **PublicSigningKeyList** ADM attribute is stored in two attributes on the computer object. The `mSMQSignCertificates` ([\[MS-ADA2\]](#) section 2.544) attribute contains an **MQUERSIGNCERTS** ([\[MS-MQMQ\]](#) section 2.2.21) structure, which contains **MQUERSIGNCERT** ([\[MS-MQMQ\]](#) section 2.2.22) structures, which in turn contain individual X.509-encoded certificates. The `mSMQDigests` ([\[MS-ADA2\]](#) section 2.511) attribute contains an array of MD5 hashes of the certificates stored in the `mSMQSignCertificates` attribute, mirroring the values of the **Digest** fields of the **MQUERSIGNCERT** structures. Each array element MUST contain the 16-byte output of the MD5 algorithm, as specified in [\[RFC1321\]](#). The certificates and digests MUST be in the same order, but there is no other sorting requirement.

The values of `mSMQSignCertificates` and `mSMQDigests` MUST be computed according to the following algorithm:

If the **PublicSigningKeyList** ADM attribute name is present in `iAttributeList` and `iDirectoryObject.PublicSigningKeyList` is populated:

- Copy the bytes in `iDirectoryObject.PublicSigningKeyList`, which is an **MQUERSIGNCERTS** structure, to `mSMQSignCertificates`.
- For each **MQUERSIGNCERT** structure in `mSMQSignCertificates`:
 - Append the bytes of the **Digest** field in the structure to `mSMQDigests`.

3.1.6.7.1.9 mSMQSettings Objects

If, as a result of this write operation, one or more of the **RoutingServer**, **DirectoryServer**, or **SupportingServer** ADM attributes is TRUE where all were previously FALSE, one or more `mSMQSettings` objects MUST be created. For each GUID in `iDirectoryObject.SiteIdentifierList`, do the following:

1. Search the Directory for a **Site** ADM element instance that has an **Identifier** ADM attribute that matches the GUID:
 1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - `iDirectoryObjectType := "Site"`
 - `iFilter :=` An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from `iDirectoryObject.SiteIdentifierList`
 - `iAttributeList :=` An array of the following **Site** ADM element attribute names:
 - **Name**
 2. If the `rStatus` returned by the Read Directory event is set to **DirectoryOperationResult.ObjectNotFound**, the current GUID MUST be skipped.
 3. If the `rStatus` returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, `rStatus` MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.

2. If any of the mSMQConfiguration attributes listed in the following table does not appear in *ConfigurationAttributeList*, a [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:

- *iPath* := *WriteDN*
- *iGuid* := *WriteGUID*
- *iAttributes* := a list of attribute names consisting of the mSMQConfiguration attributes listed in the following table that do not also appear in *ConfigurationAttributeList*

| mSMQSettings attribute | mSMQConfiguration attribute |
|---|-----------------------------|
| mSMQQMID ([MS-ADA2] section 2.534) | objectGUID |
| mSMQServices ([MS-ADA2] section 2.542) | mSMQServiceType |
| mSMQRoutingService ([MS-ADA2] section 2.540) | mSMQRoutingServices |
| mSMQDsService ([MS-ADA2] section 2.513) | mSMQDsServices |
| mSMQDependentClientService ([MS-ADA2] section 2.509) | mSMQDependentClientServices |

3. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, and processing MUST end.

4. Generate a [Create Object Using LDAP \(section 3.1.6.12\)](#) event with the following arguments:

- *iParentPath* := a distinguished name that MUST be constructed of the form specified for an mSMQSettings object in section [2.2.1](#), where "<computer name>" is the value of *iDirectoryObject.ComputerName*, and "<site name>" is the value of the *rDirectoryObject.Name* returned by the Read Directory event, with the first comma-separated element removed.
- *iChildName* := "MSMQ Settings"
- *iObjectClass* := "mSMQSettings"
- *iAttributes* := a list of name-value pairs consisting of the mSMQSettings attribute names listed in the table in step 2 and the corresponding mSMQConfiguration values copied from *ConfigurationAttributeList* or the values returned by the Get Object Properties Using LDAP event in step 3.

5. If the Create Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Create Object Using LDAP event, and processing MUST end.

If, as a result of this write operation, all of the **RoutingServer**, **DirectoryServer**, or **SupportingServer** ADM attributes are FALSE where previously one or more were TRUE, one or more mSMQSettings objects MUST be deleted. For each GUID in the **SiteIdentifierList** ADM attribute, do the following:

1. Search the Directory for a **Site** ADM element instance that has an **Identifier** ADM attribute that matches the GUID:

1. Generate a Read Directory (section 3.1.6.3) event with the following arguments:

- *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions**:
 - "Identifier" EQUALS the current GUID from *iDirectoryObject.SiteIdentifierList*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **Name**
2. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.ObjectNotFound**, the current GUID MUST be skipped.
 3. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
2. A [Delete Object Using LDAP \(section 3.1.6.13\)](#) event MUST be generated with the following arguments:
 - *iPath* := a distinguished name MUST be constructed of the form specified for an mSMQSettings object in section [2.2.1](#), where "<computer name>" is the value of *iDirectoryObject.ComputerName*, and "<site name>" is the value of the *rDirectoryObject.Name* returned by the Read Directory event.
 3. If the Delete Object Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Delete Object Using LDAP event, and processing MUST end.

If, as a result of this write operation, one or more of the **RoutingServer**, **DirectoryServer**, or **SupportingServer** ADM element attributes is TRUE, where this condition was also previously true, and one or more of those ADM attributes was written, or the **DirectoryServerType** or **RemoteAccessServer** ADM attributes were written, one or more mSMQSettings objects MUST be updated. For each GUID in the **SiteIdentifierList** ADM attribute, do the following:

1. Search the Directory for a **Site** ADM element instance that has an **Identifier** ADM attribute that matches the GUID:
 1. Generate a Read Directory event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions**:
 - "Identifier" EQUALS the current GUID from *iDirectoryObject.SiteIdentifierList*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **Name**
 2. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.ObjectNotFound**, the current GUID MUST be skipped.
 3. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.

2. A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event MUST be generated with the following arguments:
 - *iPath* := a distinguished name MUST be constructed of the form specified for an mSMQSettings object in section [2.2.1](#), where "<computer name>" is the value of **QueueManager.ComputerName**, and "<site name>" is the value of the *rDirectoryObject.Name* returned by the Read Directory event.
 - *iAttributes* := a list of attribute name-value pairs consisting of the mSMQSettings attributes listed in the preceding table in this section for which the corresponding mSMQConfiguration attributes appear in *ConfigurationAttributeList* and the values of those mSMQConfiguration attributes.
3. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing MUST end.

3.1.6.7.1.10 PublicSigningKeyList

The **PublicSigningKeyList** ADM attribute is stored in two attributes on the computer object, as specified in section [3.1.6.11.1](#). When writing the **Certificates** ADM attribute, both mSMQSignCertificates and mSMQDigests MUST be written. In addition, the **MQUSERSIGNCERTS** ([\[MS-MQMQ\]](#) section 2.2.21) structure contained in the mSMQSignCertificates attribute can be written only atomically; so to add, delete, or write an individual certificate, the mSMQSignCertificates attribute MUST be read first; the **MQUSERSIGNCERTS** structure modified (by adding, deleting, or writing a **MQUSERSIGNCERT** structure within it); then the mSMQSignCertificates attribute written with the modified structure. The appropriate operation (add, delete, or write) MUST be performed on the corresponding MD5 digest in the mSMQDigests attribute. The certificates and the digests MUST be in the same order.

3.1.6.7.2 Queue

3.1.6.7.2.1 Preconditions

The arguments supplied to the [Write Directory \(section 3.1.6.7\)](#) event MUST meet the following requirement:

- At least one of the *iDirectoryObject.FullPath*, *iDirectoryObject.DirectoryPath*, *iDirectoryObject.Identifier*, or *iDirectoryObject.Pathname* ADM attributes MUST be populated.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing of the event MUST end.

3.1.6.7.2.2 Write

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **Queue** ADM element attributes listed in section [3.1.6.20.2](#).
2. Let *QueueAttributeList* be a list of mSMQQueue attribute names, initialized to be empty. For each mSMQQueue attribute listed in the following table, if the corresponding **Queue** ADM element attribute appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the mSMQQueue attribute name to *QueueAttributeList* and compute the value for the attribute as shown. **Queue** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored in this step.

| mSMQQueue attribute | Attribute value computation |
|---|---|
| mSMQLabelEx ([MS-ADA2] section 2.523) | Directory string transformed from <i>iDirectoryObject.Label</i> . |
| mSMQQueueType ([MS-ADA2] section 2.538) | GUID copied from <i>iDirectoryObject.Type</i> . |
| mSMQJournal ([MS-ADA2] section 2.520) | If <i>iDirectoryObject.Journaling</i> is TRUE, mSMQJournal is 1 Else mSMQJournal is 0. |
| mSMQQueueQuota ([MS-ADA2] section 2.537) | Integer copied from <i>iDirectoryObject.Quota</i> . |
| mSMQQueueJournalQuota ([MS-ADA2] section 2.521) | Integer copied from <i>iDirectoryObject.JournalQuota</i> . |
| mSMQAuthenticate ([MS-ADA2] section 2.503) | If <i>iDirectoryObject.Authentication</i> is TRUE, mSMQAuthenticate is 1 Else mSMQAuthenticate is 0. |
| mSMQPrivacyLevel ([MS-ADA2] section 2.533) | Computed from <i>iDirectoryObject.PrivacyLevel</i> . See section 3.1.6.7.2.4 . |
| MSMQ-MulticastAddress ([MS-ADA2] section 2.500) | Directory string transformed from <i>iDirectoryObject.MulticastAddress</i> . |
| mSMQBasePriority ([MS-ADA2] section 2.504) | Integer copied from <i>iDirectoryObject.BasePriority</i> . |

3. Let *WriteGUID* be a distinguished name, initialized to be empty. If *iDirectoryObject.Identifier* is populated, the value of *iDirectoryObject.Identifier* MUST be copied to *WriteGUID*.
4. Let *WriteDN* be a distinguished name, initialized to be empty.
5. If *iDirectoryObject.FullPath* is populated, the value of *iDirectoryObject.FullPath* MUST be copied to *WriteDN*.
6. If *WriteDN* is empty and *iDirectoryObject.DirectoryPath* is populated, the value of *WriteDN* MUST be the value of *iDirectoryObject.DirectoryPath* with the prefix "LDAP://" removed.
7. If *WriteDN* is empty and *iDirectoryObject.Pathname* is populated, the value of *WriteDN* MUST be a distinguished name for an mSMQQueue object, constructed according to the format specified in section [2.2.1](#), where "<computer name>" is the ComputerName portion of the value of *iDirectoryObject.Pathname* and "<queue name>" is the QueueName portion of the value of *iDirectoryObject.Pathname*, as specified in [\[MS-MQMQ\]](#) section 2.1.1, and "<queue name>" is further modified as specified in section [3.1.6.7.2.5](#).
8. A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event MUST be generated with the following arguments:
 - *iPath* := *WriteDN*
 - *iGuid* := *WriteGUID*

- *iAttributes* := a list of name-value pairs consisting of the attribute names in *QueueAttributeList* and the corresponding values, as computed in step 2.
9. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing MUST end.

3.1.6.7.2.3 Postprocessing

1. If *iAttributeList* contains the name "Security" and the *iDirectoryObject.Security* ADM attribute is populated, a [Set Object Security Using LDAP \(section 3.1.6.17\)](#) event MUST be generated with the following arguments:
 - *iPath* := *WriteDN*
 - *iGuid* := *WriteGUID*
 - *iSecurity* := *iDirectoryObject.Security*
2. If the Set Object Security Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Security Using LDAP event, and processing MUST end.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.7.2.4 PrivacyLevel

The *mSMQPrivacyLevel* attribute is an integer that MUST have one of the values listed in the following table. The **PrivacyLevel** ADM attribute is an enumeration. The values MUST be converted according to the following table.

| PrivacyLevel ADM attribute value | mSMQPrivacyLevel value |
|----------------------------------|------------------------|
| None | 0 |
| Optional | 1 |
| Body | 2 |

3.1.6.7.2.5 <queue name>

If the length of the "<queue name>" does not exceed 63 characters, it MUST be used unmodified. Otherwise, it is modified according to the following algorithm:

1. Let *QName* be a Unicode string, initialized to be empty.
2. The first 55 characters of the "<queue name>" MUST be copied to *QName*.
3. A nine-character extension MUST be added to the end of *QName*. *QName* MUST be exactly 64 characters long after adding the extension. The extension consists of a Unicode dash ('-') followed by an eight-character hash string calculated from the values of the Unicode characters in the "<queue name>" as specified in section [2.2.5](#).
4. The original "<queue name>" MUST be replaced with the value of *QName*.

3.1.6.7.3 Enterprise

3.1.6.7.3.1 Preconditions

There are no preconditions for the [Write Directory \(section 3.1.6.7\)](#) event for **Enterprise** ADM element objects.

3.1.6.7.3.2 Write

1. If *iAttributeList* was not provided as an argument, then an *iAttributeList* MUST be constructed that MUST contain the names of all **Enterprise** ADM element attributes listed in section [3.1.6.20.3](#).
2. Let *EnterpriseAttributeList* be a list of mSMQEnterpriseSettings attribute names, initialized to be empty. For each mSMQEnterpriseSettings attribute listed in the following table, if the corresponding **Enterprise** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the mSMQEnterpriseSettings attribute name to *EnterpriseAttributeList* and compute the value for the attribute as shown. **Enterprise** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored in this step.

| mSMQEnterpriseSettings attribute | Attribute value computation |
|---|---|
| mSMQNameStyle ([MS-ADA2] section 2.526) | If <i>iDirectoryObject</i> . WeakenedSecurity equals disabled, mSMQNameStyle is 0x00 Else if <i>iDirectoryObject</i> . WeakenedSecurity equals enabled, mSMQNameStyle is 0x01. |
| mSMQCSPName ([MS-ADA2] section 2.508) | Computed from <i>iDirectoryObject</i> . NonLDAPCapableQueueManagerNotification . See section 3.1.6.7.3.4 . |
| mSMQLongLived ([MS-ADA2] section 2.524) | Integer copied from <i>iDirectoryObject</i> . DefaultTimeToLive . |
| mSMQVersion ([MS-ADA2] section 2.558) | If <i>iDirectoryObject</i> . OldDirectory is TRUE, mSMQVersion is 3 Else mSMQVersion is 200. |

3. A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event MUST be generated with the following arguments:
 - *iPath* := the distinguished name specified in section [2.2.1](#) for an mSMQEnterpriseSettings object.
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *EnterpriseAttributeList* and the corresponding values, as computed in step 2.
4. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing MUST end.

3.1.6.7.3.3 Postprocessing

1. If *iAttributeList* contains the name "Security" and the *iDirectoryObject.Security* ADM attribute is populated, a [Set Object Security Using LDAP \(section 3.1.6.17\)](#) event MUST be generated with the following arguments:
 - *iPath* := the distinguished name specified in section [2.2.1](#) for an mSMQEnterpriseSettings object.
 - *iSecurity* := *iDirectoryObject.Security*
2. If the Set Object Security Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Security Using LDAP event, and processing MUST end.
3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.7.3.4 mSMQCSPName

The mSMQCSPName attribute is a string that MUST have one of the values listed in the following table. The **NonLDAPCapableQueueManagerNotification** ADM attribute is a Boolean. The values MUST be converted according to the following table.

| NonLDAPCapableQueueManagerNotification ADM attribute value | mSMQCSPName value |
|--|-------------------|
| true | "Y" |
| false | "N" |

3.1.6.7.4 Site

3.1.6.7.4.1 Preconditions

The arguments supplied to the [Write Directory \(section 3.1.6.7\)](#) event MUST meet these requirements:

- If *iAttributeList* is provided, it MUST NOT contain the names **PrimarySiteController** or **PublicSigningKeyList**. These attributes are not supported in Active Directory-based environments.
- At least one of the *iDirectoryObject.Identifier*, *iDirectoryObject.Name*, or *iDirectoryObject.FullPath* ADM attributes MUST be populated.

If any of these conditions is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.7.4.2 Write

1. If *iAttributeList* was not provided as an argument, then an *iAttributeList* MUST be constructed that MUST contain the names of all **Site** ADM element attributes listed in section [3.1.6.20.4](#), except **PrimarySiteController** and **PublicSigningKeyList**.
2. Let *SiteAttributeList* be a list of site attribute names, initialized to be empty. For each site attribute listed in the following table, if the corresponding **Site** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the site attribute

name to *SiteAttributeList* and compute the value for the attribute as shown. **Site** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or subsections referenced by the table **MUST** be ignored in this step.

| site attribute | Attribute value computation |
|---|--|
| mSMQInterval1 ([MS-ADA2] section 2.518) | Integer copied from <i>iDirectoryObject.IntraSiteReplicationInterval</i> . |
| mSMQInterval2 ([MS-ADA2] section 2.519) | Integer copied from <i>iDirectoryObject.InterSiteReplicationInterval</i> . |
| mSMQSiteForeign ([MS-ADA2] section 2.549) | If <i>iDirectoryObject.ForeignSite</i> is TRUE, mSMQSiteForeign is 0x01 Else mSMQSiteForeign is 0x00. |
| mSMQnt4Stub ([MS-ADA2] section 2.528) | If <i>iDirectoryObject.MigratedFromMsmq10</i> is TRUE, mSMQnt4Stub is 0x01 Else mSMQnt4Stub is 0x00. |

3. Let *WriteGUID* be a distinguished name, initialized to be empty. If *iDirectoryObject.Identifier* is populated, the value of *iDirectoryObject.Identifier* **MUST** be copied to *WriteGUID*.
4. Let *WriteDN* be a distinguished name, initialized to be empty.
5. If *iDirectoryObject.FullPath* is populated, the value of *iDirectoryObject.FullPath* **MUST** be copied to *WriteDN*.
6. If *WriteDN* is empty and *iDirectoryObject.Name* is populated, the value of *WriteDN* **MUST** be a distinguished name for a site object, constructed according to the format specified in section [2.2.1](#), where "<site name>" is the value of *iDirectoryObject.Name*.
7. A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event **MUST** be generated with the following arguments:
 - *iPath* := *WriteDN*
 - *iGuid* := *WriteGUID*
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *SiteAttributeList* and the corresponding values, as computed in step 2.
8. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* **MUST** be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing **MUST** end.

3.1.6.7.4.3 Postprocessing

1. If *iAttributeList* contains the name "Security" and the *iDirectoryObject.Security* attribute is populated, a [Set Object Security Using LDAP \(section 3.1.6.17\)](#) event **MUST** be generated with the following arguments:
 - *iPath* := *WriteDN*

- *iGuid* := *WriteGUID*
 - *iSecurity* := *iDirectoryObject.Security*
2. If the Set Object Security Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Security Using LDAP event, and processing MUST end.
 3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.7.5 RoutingLink

3.1.6.7.5.1 Preconditions

The arguments supplied to the [Write Directory \(section 3.1.6.7\)](#) event MUST meet the following requirement:

- At least one of the *iDirectoryObject.FullPath* or *iDirectoryObject.Identifier* ADM attributes MUST be populated.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.7.5.2 Write

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **RoutingLink** ADM element attributes listed in section [3.1.6.20.5](#).
2. Let *SiteLinkAttributeList* be a list of mSMQSiteLink attribute names, initialized to be empty. For each mSMQSiteLink attribute listed in the following table, if the corresponding **RoutingLink** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the mSMQSiteLink attribute name to *SiteLinkAttributeList* and compute the value for the attribute as shown. **RoutingLink** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored.

| mSMQSiteLink attribute | Attribute value computation |
|--|--|
| description ([MS-ADA1] section 2.153) | Directory string transformed from <i>iDirectoryObject.Description</i> . |
| mSMQCost ([MS-ADA2] section 2.507) | Integer copied from <i>iDirectoryObject.ActualCost</i> . |
| mSMQSite1 ([MS-ADA2] section 2.547) | Computed from <i>iDirectoryObject.Site1Identifier</i> . See section 3.1.6.7.5.4 . |
| mSMQSite2 ([MS-ADA2] section 2.548) | Computed from <i>iDirectoryObject.Site2Identifier</i> . See section 3.1.6.7.5.5 . |
| mSMQSiteGates ([MS-ADA2] section 2.550) | Computed from <i>iDirectoryObject.SiteGateIdentifierList</i> . See section 3.1.6.7.5.6 . |

3. Let *WriteGUID* be a distinguished name, initialized to be empty. If *iDirectoryObject.Identifier* is populated, the value of *iDirectoryObject.Identifier* MUST be copied to *WriteGUID*.
4. Let *WriteDN* be a distinguished name, initialized to be empty.

5. If *iDirectoryObject.FullPath* is populated, the value of *iDirectoryObject.FullPath* MUST be copied to *WriteDN*.
6. A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event MUST be generated with the following arguments:
 - *iPath* := *WriteDN*
 - *iGuid* := *WriteGUID*
 - *iAttributes* := a list of name-value pairs consisting of the attribute names in *SiteLinkAttributeList* and the corresponding values, as computed in step 2.
7. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing MUST end.
8. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.7.5.3 Postprocessing

None.

3.1.6.7.5.4 mSMQSite1

The value of the mSMQSite1 attribute MUST be computed according to the following algorithm:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS *iDirectoryObject.Site1Identifier*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **FullPath**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing MUST end.
3. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of mSMQSite1 MUST be the value of the *rDirectoryObject.FullPath* returned by the Read Directory event.

3.1.6.7.5.5 mSMQSite2

The value of the mSMQSite2 attribute MUST be computed according to the following algorithm:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"

- *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS *iDirectoryObject.Site2Identifier*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **FullPath**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing MUST end.
 3. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of *mSMQSite2* MUST be the value of the *rDirectoryObject.FullPath* returned by the Read Directory event.

3.1.6.7.5.6 mSMQSiteGates

The value of the *mSMQSiteGates* attribute MUST be computed according to the following algorithm:

1. Let *DNList* be a list of distinguished names, initialized to be empty.
2. For each GUID in *iDirectoryObject.SiteGateIdentifierList*, these steps MUST be performed:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "Identifier" EQUALS the current GUID from *iDirectoryObject.SiteGateIdentifierList*
 - *iAttributeList* := An array of the following **QueueManager** ADM element attribute names:
 - **FullPath**
 - If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**, this GUID MUST be skipped.
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of the *rDirectoryObject.FullPath* returned by the Read Directory event must be added to *DNList*.
3. The value of *mSMQSiteGates* MUST be the value of *DNList*.

3.1.6.7.6 User

3.1.6.7.6.1 Preconditions

The arguments supplied to the [Write Directory \(section 3.1.6.7\)](#) event MUST meet the following requirement:

- At least one of the *iDirectoryObject.Identifier* or *iDirectoryObject.SecurityIdentifier* ADM attributes MUST be populated.

If this condition is violated, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rObjectGUID* is undefined, and processing of the event MUST end.

3.1.6.7.6.2 Write

1. If *iAttributeList* was not provided as an argument, an *iAttributeList* MUST be constructed that MUST contain the names of all **User** ADM element attributes listed in section [3.1.6.20.6](#).
2. Let *UserAttributeList* be a list of user attribute names, initialized to be empty. For each user attribute listed in the following table, if the corresponding **User** ADM element attribute name appears in *iAttributeList* and that attribute is populated in *iDirectoryObject*, add the user attribute to *UserAttributeList* and compute the value for the attribute as shown. **User** ADM element attribute names that appear in *iAttributeList* but do not appear in the following table or in subsections referenced by the table MUST be ignored.

| User ADM element attribute | Attribute value computation |
|---|--|
| mSMQSignCertificates ([MS-ADA2] section 2.544) | MQUERSIGNCERTS structure ([MS-MQMQ] section 2.2.21) copied from <i>iDirectoryObject.Certificates</i> . |
| mSMQDigests ([MS-ADA2] section 2.511) | List of GUIDs copied from <i>iDirectoryObject.CertificateDigestList</i> . |

3. Let *WriteGUID* be a distinguished name, initialized to be empty. If *iDirectoryObject.Identifier* is populated, the value of *iDirectoryObject.Identifier* MUST be copied to *WriteGUID*.
4. Let *WriteDN* be a distinguished name, initialized to be empty.
5. If *WriteGUID* is empty, search the directory for the user object with that **SecurityIdentifier**:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "User"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "SecurityIdentifier" EQUALS *iDirectoryObject.SecurityIdentifier*
 - *iAttributeList* := An array of the following **User** ADM element attribute names:
 - **FullPath**
 - If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**, the value of *rDirectoryObject.FullPath* MUST be copied to *WriteDN*.
 - If the *rStatus* returned by the Read Directory event is set to any other value, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
6. A [Set Object Properties Using LDAP \(section 3.1.6.16\)](#) event MUST be generated with the following arguments:
 - *iPath* := *WriteDN*
 - *iGuid* := *WriteGUID*

- *iAttributes* := a list of name-value pairs consisting of the attribute names in *UserAttributeList* and the corresponding values, as computed in step 2.
7. If the Set Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Set Object Properties Using LDAP event, and processing MUST end.
 8. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.7.6.3 Postprocessing

None.

3.1.6.7.6.4 Note on mSMQSignCertificates and mSMQDigests

The mSMQDigests attribute holds a list of MD5 digests that correspond to the **Digest** fields of the **MQUSERSIGNCERT** ([\[MS-MQMQ\]](#) section 2.2.22) structures within the **MQUSERSIGNCERTS** ([\[MS-MQMQ\]](#) section 2.2.21) structure found in the mSMQSignCertificates attribute. The digests in mSMQDigests MUST be in the same order as the certificates in mSMQSignCertificates. Therefore, when modifying one of these attributes, the other MUST be modified at the same time. This algorithm does not enforce that requirement; it is up to implementers of this algorithm to ensure that it is met.

3.1.6.8 Resolve Queue Alias

This event MUST be generated with the following arguments:

- *iFullPath*: A Distinguished Name (DN) that can be used to lookup a directory object of type msMQ-Custom-Recipient ([\[MS-ADSC\]](#) section 2.155).

Return Values

- *rStatus*: A status code that indicates success or failure.
- *rFormatName*: The msMQ-Recipient-FormatName ([\[MS-ADA2\]](#) section 2.501) attribute of the msMQ-Custom-Recipient object. This value is defined only if *rStatus* has a value of **DirectoryOperationResult.Success**.

The algorithm MUST perform the following processing steps to resolve a queue alias:

- A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iPath* := *iFullPath*
 - *iAttributes* := a list of attribute names consisting of one element, "msMQ-Recipient-FormatName"
- If the *rStatus* returned by the Get Object Properties Using LDAP event is **DirectoryOperationResult.Success**, set *rFormatName* to the value returned in *rValues* for the msMQ-Recipient-FormatName attribute.
- Set *rStatus* to the *rStatus* returned by the Get Object Properties Using LDAP event. Processing MUST end.

3.1.6.9 Resolve Distribution List

This event MUST be generated with the following arguments:

- *iDLFormatName*: A distribution list format name as specified in [\[MS-MQMQ\]](#) section 2.1.5.

Return Values

- *rStatus*: A status code that indicates success or failure.
- *rFormatNameCollection*: A list of queue format names. This value is defined only if *rStatus* has a value of **DirectoryOperationResult.Success**.

The algorithm MUST perform the following processing steps to resolve a queue alias:

1. Instantiate a new list referred to as *newDistinguishedNameCollection*.
2. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iGuid* := the DistributionListGuid portion of *iDLFormatName*, as specified in [\[MS-MQMQ\]](#) section 2.1.5
 - *iAttributes* := a list of attribute names consisting of one element, "member"
3. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rFormatNameCollection* is undefined, and processing MUST end.
4. Copy the values returned by the Get Object Properties Using LDAP event for the member attribute into *newDistinguishedNameCollection*.
5. For each distinguished name *dName* in *newDistinguishedNameCollection*:
 1. Generate a Get Object Properties Using LDAP (section 3.1.6.14) event with the following arguments:
 - *iPath* := *dName*
 - *iAttributes* := a list of attribute names consisting of two elements, "objectClass" and "objectGuid"
 2. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rFormatNameCollection* is undefined, and processing MUST end.
 3. Let *LoopGuid* be a GUID variable, initialized to the value returned by the Get Object Properties Using LDAP event in *rValues* for the objectGuid attribute.
 4. If the value returned by the Get Object Properties Using LDAP event in *rValues* for the objectClass attribute is "mSMQQueue", construct a public format name, as specified in [\[MS-MQMQ\]](#) section 2.1.3, where *QueueGuid* is the value of *LoopGuid*, and add the format name to *rFormatNameCollection*.
 5. Else if the value returned by the Get Object Properties Using LDAP event in *rValues* for the objectClass attribute is "mSMQ-Custom-Recipient", perform the following steps:

1. Generate a [Resolve Queue Alias \(section 3.1.6.8\)](#) event with the following arguments:
 - *iFullPath* := *dName*
2. If the *rStatus* returned by the Resolve Queue Alias event is **DirectoryOperationResult.Success**, add the returned *rFormatName* to *rFormatNameCollection*.
3. Else *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
6. Else if the value returned by the Get Object Properties Using LDAP event in *rValues* for the objectClass attribute is "group", perform the following steps:
 1. Recursively generate a Resolve Distribution List event with the following arguments:
 - *iDLFormatName* := a distribution list format name constructed as specified in [\[MS-MQMQ\]](#) section 2.1.5, where *DistributionListGuid* is the value of *LoopGuid*
 2. If the *rStatus* returned by the Resolve DistributionList event is **DirectoryOperationResult.Success**, add the *rFormatNameCollection* returned by the recursive Resolve Distribution List event to *rFormatNameCollection*.
 3. Else *rStatus* MUST be set to the *rStatus* returned by the recursive Resolve Distribution List event, and processing MUST end.
7. Else *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
6. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.10 Create LDAP Attribute List

This event MUST be generated with the following arguments:

- *iDirectoryObjectType*: A string that specifies the name of the sub-type of **DirectoryObject** ADM element.
- *iADMAtributeList*: An array of strings containing the subset of the names of the directory attributes of the *iDirectoryObjectType* to be converted to the names of the LDAP attributes necessary to populate those directory attributes on an ADM element of type *iDirectoryObjectType*.

Return Values:

- *rLDAPAttributeList*: A list of name pairs, consisting of the directory attribute name and the corresponding LDAP attribute name.

The algorithm MUST perform the following actions to process this event:

- Based on *iDirectoryObjectType*, perform the conversion steps for that type of ADM element, as specified in the following sections.
 - [QueueManager \(section 3.1.6.10.1\)](#)
 - [Queue \(section 3.1.6.10.2\)](#)
 - [Enterprise \(section 3.1.6.10.3\)](#)

- [Site \(section 3.1.6.10.4\)](#)
- [RoutingLink \(section 3.1.6.10.5\)](#)
- [User \(section 3.1.6.10.6\)](#)

3.1.6.10.1 QueueManager

For each **QueueManager** ADM element attribute listed in the following table that appears in the *iADMAAttributeList* argument, add the ADM element attribute name paired with the corresponding mSMQConfiguration attribute name to *rLDAPAttributeList*. The **QueueManager** ADM element attributes that appear in the *iADMAAttributeList* argument but not in the following table MUST be ignored.

| QueueManager ADM element attribute | mSMQConfiguration attribute |
|---------------------------------------|--|
| Identifier | objectGUID ([MS-ADA3] section 2.44) |
| ComputerName | distinguishedName ([MS-ADA1] section 2.177) |
| QueueManagerVersion | mSMQComputerTypeEx ([MS-ADA2] section 2.506) |
| OperatingSystemType | mSMQOSType ([MS-ADA2] section 2.529) |
| CreateTime | whenCreated ([MS-ADA3] section 2.371) |
| ModifyTime | whenChanged ([MS-ADA3] section 2.370) |
| QueueManagerQuota | mSMQQuota ([MS-ADA2] section 2.539) |
| JournalQuota | mSMQJournalQuota ([MS-ADA2] section 2.521) |
| ForeignSystem | mSMQForeign ([MS-ADA2] section 2.516) |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) |
| SiteIdentifierList | mSMQSites ([MS-ADA2] section 2.555) |
| OutRoutingServerIdentifierList | mSMQOutRoutingServers ([MS-ADA2] section 2.530) |
| InRoutingServerIdentifierList | mSMQInRoutingServers ([MS-ADA2] section 2.517) |
| RoutingServer | mSMQRoutingServices ([MS-ADA2] section 2.541)) |
| DirectoryServer | mSMQDsServices ([MS-ADA2] section 2.514) |
| DirectoryServerType | mSMQDsServices ([MS-ADA2] section 2.514) and mSMQServiceType ([MS-ADA2] section 2.543) |
| RemoteAccessServer | mSMQServiceType ([MS-ADA2] section 2.543) |
| SupportingServer | mSMQDependentClientServices ([MS-ADA2] section 2.510) |
| PublicEncryptionKeyList | mSMQEncryptKey ([MS-ADA2] section 2.515) |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) |

3.1.6.10.2 Queue

For each **Queue** ADM element attribute listed in the following table that appears in the *iADMAAttributeList* argument, add the ADM element attribute name paired with the corresponding mSMQConfiguration attribute name to *rLDAPAttributeList*. The **Queue** ADM element attributes that appear in the *iADMAAttributeList* argument but not in the following table MUST be ignored.

| Queue ADM element attribute | mSMQQueue attribute |
|-----------------------------|---|
| Identifier | objectGUID ([MS-ADA3] section 2.44) |
| Label | mSMQLabelEx ([MS-ADA2] section 2.523) |
| CreateTime | whenCreated ([MS-ADA3] section 2.371) |
| ModifyTime | whenChanged ([MS-ADA3] section 2.370) |
| Type | mSMQQueueType ([MS-ADA2] section 2.538) |
| Pathname | distinguishedName ([MS-ADA1] section 2.177), mSMQQueueNameExt ([MS-ADA2] section 2.536) |
| QualifiedPathname | distinguishedName ([MS-ADA1] section 2.177), mSMQQueueNameExt ([MS-ADA2] section 2.536) |
| Journaling | mSMQJournal ([MS-ADA2] section 2.520) |
| Quota | mSMQQueueQuota ([MS-ADA2] section 2.537) |
| JournalQuota | mSMQQueueJournalQuota ([MS-ADA2] section 2.521) |
| Authentication | mSMQAuthenticate ([MS-ADA2] section 2.503) |
| PrivacyLevel | mSMQPrivacyLevel ([MS-ADA2] section 2.533) |
| Transactional | mSMQTransactional ([MS-ADA2] section 2.556) |
| MulticastAddress | MSMQ-MulticastAddress ([MS-ADA2] section 2.500) |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) |
| BasePriority | mSMQBasePriority ([MS-ADA2] section 2.504) |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) |
| DirectoryPath | distinguishedName ([MS-ADA1] section 2.177) |

3.1.6.10.3 Enterprise

For each **Enterprise** ADM element attribute listed in the following table that appears in the *iADMAAttributeList* argument, add the ADM element attribute name paired with the corresponding mSMQConfiguration attribute name to *rLDAPAttributeList*. The **Enterprise** ADM element attributes that appear in the *iADMAAttributeList* argument but not in the following table MUST be ignored.

| Enterprise ADM element attribute | mSMQEnterpriseSettings attribute |
|----------------------------------|-------------------------------------|
| Identifier | objectGUID ([MS-ADA3] section 2.44) |

| Enterprise ADM element attribute | mSMQEnterpriseSettings attribute |
|---|---|
| Name | None |
| WeakenedSecurity | mSMQNameStyle ([MS-ADA2] section 2.526) |
| NonLDAPCapableQueueManagerNotification | mSMQCSPName ([MS-ADA2] section 2.508) |
| DefaultTimeToLive | mSMQLongLived ([MS-ADA2] section 2.524) |
| OldDirectory | mSMQVersion ([MS-ADA2] section 2.558) |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) |

3.1.6.10.4 Site

For each **Site** ADM element attribute listed in the following table that appears in the *iADMAAttributeList* argument, add the ADM element attribute name paired with the corresponding mSMQConfiguration attribute name to *rLDAPAttributeList*. The **Site** ADM element attributes that appear in the *iADMAAttributeList* argument but not in the following table MUST be ignored.

| Site ADM element attribute | site attribute |
|-------------------------------------|---|
| Identifier | objectGUID ([MS-ADA3] section 2.44) |
| Name | cn ([MS-ADA1] section 2.110) |
| IntraSiteReplicationInterval | mSMQInterval1 ([MS-ADA2] section 2.518) |
| InterSiteReplicationInterval | mSMQInterval2 ([MS-ADA2] section 2.519) |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) |
| Foreign | mSMQSiteForeign ([MS-ADA2] section 2.549) |
| MigratedFromMsmq10 | mSMQNt4Stub ([MS-ADA2] section 2.528) |
| Security | nTSecurityDescriptor ([MS-ADA3] section 2.37) |

3.1.6.10.5 RoutingLink

For each **RoutingLink** ADM element attribute listed in the following table that appears in the *iADMAAttributeList* argument, add the ADM element attribute name paired with the corresponding mSMQConfiguration attribute name to *rLDAPAttributeList*. The **RoutingLink** ADM element attributes that appear in the *iADMAAttributeList* argument but not in the following table MUST be ignored.

| RoutingLink ADM element attribute | mSMQSiteLink attribute |
|-----------------------------------|---|
| Identifier | objectGUID ([MS-ADA3] section 2.44) |
| Description | description ([MS-ADA1] section 2.153) |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) |
| ActualCost | mSMQCost ([MS-ADA2] section 2.507) |
| Site1Identifier | mSMQSite1 ([MS-ADA2] section 2.547) |

| RoutingLink ADM element attribute | mSMQSiteLink attribute |
|-----------------------------------|---|
| Site2Identifier | mSMQSite2 ([MS-ADA2] section 2.548) |
| SiteGateIdentifierList | mSMQSiteGates ([MS-ADA2] section 2.550) |

3.1.6.10.6 User

For each **User** ADM element attribute listed in the following table that appears in the *iADMAAttributeList* argument, add the ADM element attribute name paired with the corresponding mSMQConfiguration attribute name to *rLDAPAttributeList*. The **User** ADM element attributes that appear in the *iADMAAttributeList* argument but not in the following table MUST be ignored.

| User ADM element attribute | user attribute |
|----------------------------|--|
| Identifier | objectGUID ([MS-ADA3] section 2.44) |
| SecurityIdentifier | objectSid ([MS-ADA3] section 2.45) |
| CertificateDigestList | mSMQDigests ([MS-ADA2] section 2.511) |
| Certificates | mSMQSignCertificates ([MS-ADA2] section 2.544) |
| FullPath | distinguishedName ([MS-ADA1] section 2.177) |

3.1.6.11 Create ADM Element From LDAP Values

This event MUST be generated with the following arguments:

- *iDirectoryObjectType*: A string that specifies the name of the sub-type of the **DirectoryObject** ADM element to be created.
- *iADMAAttributeList*: An array of ADM element attribute names indicating which attributes to set on the created ADM element.
- *iLDAPAttributeList*: An array of LDAP attribute names.
- *iValues*: An array of LDAP attribute values, in the same order as the names in *iLDAPAttributeList*.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this operation.
- *rDirectoryObject*: The created ADM element instance. This value is undefined if *rStatus* is not **DirectoryOperationResult.Success**.

3.1.6.11.1 QueueManager

1. A **QueueManager** ADM element instance MUST be created, and *rDirectoryObject* MUST be set to this new **QueueManager** ADM element instance.
2. The attributes listed in the *iADMAAttributeList* argument MUST be set on *rDirectoryObject*, using the attribute names in the *iLDAPAttributeList* argument and the corresponding values in the *iValues* argument, according to the following table. If the value of the LDAP attribute required to compute the **QueueManager** ADM element attribute is empty in the *iValues* argument, depending on the attribute, either this condition is an error, or a default implementation-specific

value SHOULD<10> be supplied. If an error condition occurs, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.

| QueueManager ADM element attribute | Attribute value computation | If not set, default value or error? |
|---------------------------------------|--|--|
| Identifier | GUID copied from objectGUID ([MS-ADA3] section 2.44). | Error |
| ComputerName | Computed from distinguishedName ([MS-ADA1] section 2.177); see section 3.1.6.11.1.1 . | Error |
| QueueManagerVersion | Unicode string transformed from mSMQComputerTypeEx ([MS-ADA2] section 2.506). | Empty string |
| OperatingSystemType | Computed from mSMQOSType ([MS-ADA2] section 2.529); see section 3.1.6.11.1.2 . | The enumeration value Unknown . |
| QualifiedComputerName | Set in a following step. | N/A |
| CreateTime | Integer time value expressed as the number of seconds elapsed from midnight (00:00:00), January 1, 1970 UTC to whenCreated ([MS-ADA3] section 2.371). | Error |
| ModifyTime | Integer time value expressed as the number of seconds elapsed from midnight (00:00:00), January 1, 1970 UTC to whenChanged ([MS-ADA3] section 2.370). | Error |
| QueueManagerQuota | Integer copied from mSMQQuota ([MS-ADA2] section 2.539). | 0x00100000 |
| JournalQuota | Integer copied from mSMQJournalQuota ([MS-ADA2] section 2.521). | 0xFFFFFFFF |
| ForeignSystem | If mSMQForeign ([MS-ADA2] section 2.516) equals 0x01, ForeignSystem is TRUE Else if mSMQForeign equals 0x00, ForeignSystem is FALSE. | FALSE |
| FullPath | Distinguished name copied from distinguishedName ([MS-ADA1] section 2.177). | Error |
| SiteIdentifierList | List of GUIDs copied from mSMQSites ([MS-ADA2] section 2.555). | Empty list |
| OutRoutingServerIdentifierList | Computed from mSMQOutRoutingServers ([MS- | Empty list |

| QueueManager ADM element attribute | Attribute value computation | If not set, default value or error? |
|--------------------------------------|--|--|
| | ADA2 section 2.530); see section 3.1.6.11.1.3 . | |
| InRoutingServerIdentifierList | Computed from mSMQInRoutingServers ([MS-ADA2] section 2.517); see section 3.1.6.11.1.4 . | Empty list |
| RoutingServer | If mSMQRoutingServices ([MS-ADA2] section 2.541) equals 0x01, RoutingServer is TRUE Else if mSMQRoutingServices equals 0x00, RoutingServer is FALSE. | FALSE |
| DirectoryServer | If mSMQDsServices ([MS-ADA2] section 2.514) equals 0x01, DirectoryServer is TRUE Else if mSMQDsServices equals 0x00, DirectoryServer is FALSE. | FALSE |
| DirectoryServerType | Computed from mSMQDsServices ([MS-ADA2] section 2.514) and mSMQServiceType ([MS-ADA2] section 2.543); see section 3.1.6.11.1.5 . | The enumeration value Standalone if DirectoryServer equals TRUE; undefined if DirectoryServer equals FALSE or is unpopulated. |
| RemoteAccessServer | If bit 0x00000010 of mSMQServiceType ([MS-ADA2] section 2.543) is set, RemoteAccessServer is TRUE Else if bit 0x00000010 of mSMQServiceType is not set, RemoteAccessServer is FALSE. | FALSE |
| SupportingServer | If mSMQDependentClientServices ([MS-ADA2] section 2.510) equals 0x01, SupportingServer is TRUE Else if mSMQDependentClientServices equals 0x00, SupportingServer is FALSE. | FALSE |
| PublicEncryptionKeyList | MQDSPUBLICKEYS structure ([MS-MQMQ] section 2.2.2) copied from mSMQEncryptKey ([MS-ADA2] section 2.515). | Empty list |

| QueueManager ADM element attribute | Attribute value computation | If not set, default value or error? |
|------------------------------------|--|-------------------------------------|
| PublicSigningKeyList | Set in a following step. | N/A |
| Security | Security descriptor copied from nTSecurityDescriptor ([MS-ADA3] section 2.37). | Error |
| Clustered | Set in a following step. | N/A |

3. If none of the **QueueManager** ADM element attribute names **QualifiedComputerName**, **PublicSigningKeyList**, **OperatingSystemVersion**, or **Clustered** appears in the *iADMAAttributeList* argument, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.
4. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iPath* := a distinguished name of the form for a computer object as specified in section [2.2.1](#), created by removing the first comma-separated element of the value of *rDirectoryObject.FullPath*.
 - *iAttributes* := a list of attribute names. The following table lists the computer attributes required to compute the values of the **QueueManager** ADM element attributes **QualifiedComputerName**, **PublicSigningKeyList**, **OperatingSystemVersion**, and **Clustered**. For each **QueueManager** ADM element attribute that appears in *ConfigurationAttributeList*, the required computer attribute name MUST appear in the *iAttributes* argument.

| QueueManager ADM element attribute | Required computer attribute | Attribute value computation |
|------------------------------------|--|---|
| QualifiedComputerName | dNSHostName ([MS-ADA1] section 2.185) | Unicode string transformed from dNSHostName. |
| PublicSigningKeyList | mSMQSignCertificates ([MS-ADA2] section 2.544) | MQUERSIGNCERTS ([MS-MQMQ] section 2.2.21) structure copied from mSMQSignCertificates. |
| OperatingSystemVersion | operatingSystemVersion ([MS-ADA3] section 2.56) | Unicode string transformed from operatingSystemVersion. |
| Clustered | servicePrincipalName ([MS-ADA3] section 2.253) | See section 3.1.6.11.1.6 . |

5. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, *rDirectoryObject* is undefined, and processing MUST end. If dNSHostName or servicePrincipalName is required, but the value returned in *rValues* for that attribute is empty, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end. If operatingSystemVersion is required but not set on the computer object, a default value MUST be supplied. [<11>](#)
6. For each of the **QueueManager** ADM element attributes **QualifiedComputerName**, **PublicSigningKeyList**, and **Clustered** that appears in the *iADMAAttributeList* argument, that attribute must be set on *rDirectoryObject* according to the computation rule specified in the table

in step 4. If **PublicSigningKeyList** appears in **ReadIterator.AttributeList**, but **mSMQSignCertificates** is not set on the computer object, a default value MUST be supplied for the **PublicSigningKeyList** ADM attribute. <12>

7. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.11.1.1 ComputerName

The value of *rDirectoryObject.ComputerName* MUST be computed from the value of the distinguishedName attribute. *rDirectoryObject.ComputerName* must be set to the "<computer name>" portion of the value of the distinguishedName attribute, as specified in section 2.2.1 for the distinguished name of an mSMQConfiguration object.

3.1.6.11.1.2 OperatingSystemType

The mSMQOSType attribute is an integer that MUST have one of the values listed in the following table. The **OperatingSystemType** ADM attribute is an enumeration. The values MUST be converted according to this table.

| mSMQOSType value | OperatingSystemType ADM attribute value |
|------------------|---|
| 0x00000000 | Other |
| 0x00000100 | Foreign |
| 0x00000200 | Win95 |
| 0x00000300 | WinClient |
| 0x00000400 | WinServer |
| 0x00000500 | WinEnt |

3.1.6.11.1.3 OutRoutingServerIdentifierList

The value of *rDirectoryObject.OutRoutingServerIdentifierList* MUST be computed from the value of the mSMQOutRoutingServers attribute. For each distinguished name in the mSMQOutRoutingServers attribute, these steps MUST be followed:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "FullPath" EQUALS the current distinguished name
 - *iAttributeList* := An array of the following **QueueManager** ADM element attribute names:
 - **Identifier**
2. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.ObjectNotFound**:
 - The current distinguished name MUST be skipped.

3. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**:
 - *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and *rDirectoryObject* is undefined.
 - Processing MUST end.
4. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**:
 - The GUID returned in *rDirectoryObject.Identifier* by the Read Directory event MUST be added to the value of the **OutRoutingServerIdentifierList** ADM attribute.

3.1.6.11.1.4 InRoutingServerIdentifierList

The value of *rDirectoryObject.InRoutingServerIdentifierList* MUST be computed from the value of the *mSMQInRoutingServers* attribute. For each distinguished name in the *mSMQInRoutingServers* attribute, these steps MUST be followed:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "QueueManager"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "FullPath" EQUALS the current distinguished name
 - *iAttributeList* := An array of the following **QueueManager** ADM attribute names:
 - **Identifier**
2. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.ObjectNotFound**:
 - The current distinguished name MUST be skipped.
3. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**:
 - *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and *rDirectoryObject* is undefined.
 - Processing MUST end.
4. If the *rStatus* returned by the Read Directory event is set to **DirectoryOperationResult.Success**:
 - The GUID returned in *rDirectoryObject.Identifier* by the Read Directory event MUST be added to the value of the **InRoutingServerIdentifierList** ADM attribute.

3.1.6.11.1.5 DirectoryServerType

The *rDirectoryObject.DirectoryServerType* attribute MUST NOT be set if *rDirectoryObject.DirectoryServer* is returned as FALSE or is not populated.

The `mSMQServiceType` attribute is an integer containing a bitmap. If one of the bits listed in the following table is set, `rDirectoryObject.DirectoryServerType` MUST be set to the corresponding enumeration value. If none of the bits listed in the following table is set, `rDirectoryObject.DirectoryServerType` MUST be set to the enumeration value **Standalone**.

| mSMQServiceType value | DirectoryServerType value |
|------------------------------|------------------------------------|
| 0x00000002 | BackupSiteController |
| 0x00000004 | PrimarySiteController |
| 0x00000008 | PrimaryEnterpriseController |

3.1.6.11.1.6 Clustered

The value of `rDirectoryObject.Clustered` MUST be computed from the `servicePrincipalName` attribute. If the value of the `servicePrincipalName` attribute contains the substring "MSClusterVirtualServer", the value of the **Clustered** ADM attribute MUST be TRUE; otherwise, it MUST be FALSE.

3.1.6.11.2 Queue

1. A **Queue** ADM element instance MUST be created, and `rDirectoryObject` MUST be set to this new **Queue** ADM element instance.
2. The attributes listed in the `iADMAtributeList` argument MUST be set on `rDirectoryObject`, using the attribute names in the `iLDAPAttributeList` argument and the corresponding values in the `iValues` argument, according to the following table. If the value of the LDAP attribute required to compute the **Queue** ADM element attribute is empty in the `iValues` argument, depending on the attribute, either that is an error condition, or a default value MUST be used. If an error condition occurs, `rStatus` MUST be set to **DirectoryOperationResult.GenericError**, `rDirectoryObject` is undefined, and processing MUST end.

| Queue attribute | Attribute value computation | If not set, default value or error? |
|------------------------|---|--|
| Identifier | GUID copied from <code>objectGUID</code> ([MS-ADA3] section 2.44). | Error. |
| Label | Unicode string transformed from <code>mSMQLabelEx</code> ([MS-ADA2] section 2.523). | Default value: empty Unicode string. |
| CreateTime | Integer time value expressed as the number of seconds elapsed from midnight (00:00:00), January 1, 1970 UTC to <code>whenCreated</code> ([MS-ADA3] section 2.371). | Error. |
| ModifyTime | Integer time value expressed as the number of seconds elapsed from midnight (00:00:00), January 1, 1970 UTC to <code>whenChanged</code> ([MS-ADA3] section 2.370). | Error. |
| Type | GUID copied from <code>mSMQQueueType</code> ([MS-ADA2] section 2.538). | Default value: a GUID with all fields set to zero. |

| Queue attribute | Attribute value computation | If not set, default value or error? |
|--------------------------|---|--|
| Pathname | Computed from distinguishedName ([MS-ADA1] section 2.177) and mSMQQueueNameExt ([MS-ADA2] section 2.536); see section 3.1.6.11.2.1. | See section 3.1.6.11.2.1. |
| QualifiedPathName | Set in a following step. distinguishedName ([MS-ADA1] section 2.177) and mSMQQueueNameExt ([MS-ADA2] section 2.536) are required. | N/A |
| Journaling | If mSMQJournal ([MS-ADA2] section 2.520) equals 1, Journaling is TRUE Else if mSMQJournal equals 0, Journaling is FALSE. | Default value: FALSE. |
| Quota | Integer copied from mSMQQueueQuota ([MS-ADA2] section 2.537). | Default value: 0xFFFFFFFF. |
| JournalQuota | Integer copied from mSMQQueueJournalQuota ([MS-ADA2] section 2.535). | Default value: 0xFFFFFFFF. |
| Authentication | If mSMQAuthenticate ([MS-ADA2] section 2.503) equals 1, Authentication is TRUE Else if mSMQAuthenticate equals 0, Authentication is FALSE. | Default value: FALSE. |
| PrivacyLevel | Computed from mSMQPrivacyLevel ([MS-ADA2] section 2.533); see section 3.1.6.11.2.3. | Default value: enumeration value Optional . |
| Transactional | If mSMQTransactional ([MS-ADA2] section 2.556) equals 1, Transactional is TRUE Else if mSMQTransactional equals 0, Transactional is FALSE. | Default value: FALSE. |
| MulticastAddress | Unicode string transformed from MSMQ-MulticastAddress ([MS-ADA2] section 2.500). | Default value: empty Unicode string. |
| Security | Security descriptor copied from nTSecurityDescriptor ([MS-ADA3] section 2.37). | Error |
| BasePriority | Integer copied from mSMQBasePriority ([MS-ADA2] section 2.504). | Default value: zero. |
| FullPath | Distinguished name copied from distinguishedName ([MS-ADA1] section 2.177). | Error |
| DirectoryPath | Unicode string computed from distinguishedName ([MS-ADA1] section 2.177) by transforming to Unicode and prepending the Unicode string "LDAP://". | Error |

3. If the **Queue** ADM element attribute name **QualifiedPathname** does not appear in *iADMAAttributeList*, *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.
4. A [Get Object Properties Using LDAP \(section 3.1.6.14\)](#) event MUST be generated with the following arguments:
 - *iPath* := a distinguished name of the form for a computer object as specified in section [2.2.1](#), created by removing the first two comma-separated elements of the value in the *iValues* argument for the mSMQQueue attribute distinguishedName.
 - *iAttributes* := a list of attribute names consisting of one element, "dNSHostName"
5. If the Get Object Properties Using LDAP event returns an *rStatus* that is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the *rStatus* returned by the Get Object Properties Using LDAP event, and processing MUST end. If the value returned in *rValues* for the dNSHostName attribute is empty, *rDirectoryObject.QualifiedPathname* MUST be set to an empty string. Otherwise, let *DNSname* be a string initialized to the value returned in *rValues* for the computer attribute dNSHostName.
6. *rDirectoryObject.QualifiedPathname* MUST be computed from the values in the *iValues* argument of the mSMQQueue attributes distinguishedName and mSMQQueueNameExt and the value of *DNSname*, as specified in section [3.1.6.11.2.2](#).
7. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.11.2.1 Pathname

The **Pathname** ADM attribute MUST be computed from the values in the *iValues* argument of the distinguishedName and mSMQQueueNameExt attributes according to the following algorithm:

1. Let *CompName*, *QName*, *ExtName*, and *OutputName* be Unicode strings, initialized to be empty.
2. *CompName* and *QName* MUST be the "<computer name>" and "<queue name>", respectively, extracted from the value of the distinguishedName attribute in the *iValues* argument and transformed to Unicode strings. The "<computer name>" and "<queue name>" as elements of the distinguished name for an mSMQQueue object are specified in section [2.2.1](#). If the value in the *iValues* argument of the distinguishedName attribute is empty, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
3. If the value of *QName* is 64 Unicode characters long, *ExtName* MUST be set to the value of the mSMQQueueNameExt attribute transformed to a Unicode string; then *QName* MUST be truncated by removing the last nine Unicode characters. If the value in the *iValues* argument of the mSMQQueueNameExt attribute is empty, *ExtName* MUST remain empty.
4. The value of *CompName* MUST be copied to *OutputName*.
5. A Unicode backslash character ('\') MUST be appended to *OutputName*.
6. All Unicode backslash characters in the value of *QName* MUST be removed.
7. The value of *QName* MUST be appended to the value of *OutputName*.
8. If *ExtName* is not an empty string, the value of *ExtName* MUST be appended to the value of *OutputName*.
9. All alphabetic characters in the value of *OutputName* MUST be converted to lowercase.

10. The value of *rDirectoryObject.Pathname* MUST be the value of *OutputName*.

3.1.6.11.2.2 QualifiedPathname

The **QualifiedPathname** ADM attribute MUST be computed from the values in the *iValues* argument of the *mSMQQueue* attributes *distinguishedName* and *mSMQQueueNameExt* and the value of *DNSname*:

1. Let *QName*, *ExtName*, and *OutputName* be Unicode strings, initially set to empty.
2. *QName* MUST be the "<queue name>" extracted from the *distinguishedName* attribute in the *iValues* argument and transformed to a Unicode string. The "<queue name>" as an element of the distinguished name of an *mSMQQueue* object is specified in section 2.2.1. If the value in the *iValues* argument of the *distinguishedName* attribute is empty, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
3. If the *QName* is 64 Unicode characters long, *ExtName* MUST be set to the value of the *mSMQQueueNameExt* attribute and transformed to a Unicode string; then *QName* MUST be truncated by removing the last nine Unicode characters. If the value in the *iValues* argument of the *mSMQQueueNameExt* attribute is empty, *ExtName* MUST remain empty.
4. The value of the *dNSHostName* attribute MUST be copied to *OutputName*.
5. A Unicode backslash character ('\') MUST be appended to *OutputName*.
6. The value of *QName* MUST be appended to *OutputName*.
7. If *ExtName* is not an empty string, the value of *ExtName* MUST be appended to *OutputName*.
8. All alphabetic characters in the value *OutputName* MUST be converted to lowercase.
9. The value of *rDirectoryObject.QualifiedPathname* MUST be set to the value of *OutputName*.

3.1.6.11.2.3 PrivacyLevel

The *mSMQPrivacyLevel* attribute is an integer that MUST have one of the values listed in the following table. The **PrivacyLevel** ADM attribute is an enumeration. The values MUST be converted according to this table.

| mSMQPrivacyLevel value | PrivacyLevel ADM attribute value |
|-------------------------------|---|
| 0 | None |
| 1 | Optional |
| 2 | Body |

3.1.6.11.3 Enterprise

1. An **Enterprise** ADM element instance MUST be created, and *rDirectoryObject* MUST be set to this new **Enterprise** ADM element instance.
2. The attributes listed in the *iADMAttributeList* argument MUST be set on *rDirectoryObject*, using the attribute names in the *iLDAPAttributeList* argument and the corresponding values in the *iValues* argument, according to the following table. If the value of the LDAP attribute required to compute the **Enterprise** ADM element attribute is empty in the *iValues* argument, depending on

the attribute, either a default value MUST be supplied, or that is an error condition. <13> If an error condition occurs, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.

| Enterprise ADM element attribute | Attribute value computation | If not set, default value or error? |
|---|---|-------------------------------------|
| Identifier | GUID copied from objectGUID ([MS-ADA3] section 2.44). | Error |
| Name | See section 3.1.6.11.3.1. | See section 3.1.6.11.3.1 |
| WeakenedSecurity | Computed from mSMQNameStyle ([MS-ADA2] section 2.526); see section 3.1.6.11.3.2. | Default value |
| NonLDAPCapableQueueManagerNotification | Computed from mSMQCSPName ([MS-ADA2] section 2.508); see section 3.1.6.11.3.3. | Default value |
| DefaultTimeToLive | Integer copied from mSMQLongLived ([MS-ADA2] section 2.524). | Default value |
| OldDirectory | If mSMQVersion ([MS-ADA2] section 2.558) equals 3, OldDirectory is TRUE Else if mSMQVersion equals 200, OldDirectory is FALSE. | Default value |
| Security | Security descriptor copied from nTSecurityDescriptor ([MS-ADA3] section 2.37). | Error |

3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.11.3.1 Name

The **Name** ADM attribute MUST be computed by retrieving the rootDomainNamingContext as specified in [MS-ADTS] section 3.1.1.3.2.16. If the rootDomainNamingContext cannot be retrieved, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.

3.1.6.11.3.2 WeakenedSecurity

The mSMQNameStyle attribute is an integer that MUST have one of the values listed in the following table. The **WeakenedSecurity** ADM attribute is a Boolean. The values MUST be converted according to this table.

| mSMQNameStyle value | WeakenedSecurity ADM attribute value |
|---------------------|--------------------------------------|
| 0x00 | False |
| 0x01 | True |
| 0x02 | Use internal default. <14> |

3.1.6.11.3.3 NonLDAPCapableQueueManagerNotification

The mSMQCSPName attribute is a string that MUST have one of the values listed in the following table. The **NonLDAPCapableQueueManagerNotification** ADM attribute is a Boolean. The values MUST be converted according to this table.

| mSMQCSPName value | NonLDAPCapableQueueManagerNotification ADM attribute value |
|-------------------|--|
| "Y" | True |
| "N" | False |

3.1.6.11.4 Site

1. A **Site** ADM element instance MUST be created, and *rDirectoryObject* MUST be set to this new **Site** ADM element instance.
2. The attributes listed in the *iADMAtributeList* argument MUST be set on *rDirectoryObject*, using the attribute names in the *iLDAPAttributeList* argument and the corresponding values in the *iValues* argument, according to the following table. If the value of the LDAP attribute required to compute the **Site** ADM element attribute is empty in the *iValues* argument, depending on the attribute, either a default value MUST be supplied, or that is an error condition. <15> If an error condition occurs, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.

| Site ADM element attribute | Attribute value computation | If not set, default value or error? |
|-------------------------------------|--|-------------------------------------|
| Identifier | GUID copied from objectGUID ([MS-ADA3] section 2.44). | Error |
| Name | Unicode string transformed from cn ([MS-ADA1] section 2.110). | Error |
| IntraSiteReplicationInterval | Integer copied from mSMQInterval1 ([MS-ADA2] section 2.518). | Default value |
| InterSiteReplicationInterval | Integer copied from mSMQInterval2 ([MS-ADA2] section 2.519). | Default value |
| FullPath | Distinguished name copied from distinguishedName ([MS-ADA1] section 2.177). | Error |
| Foreign | If mSMQSiteForeign ([MS-ADA2] section 2.549) equals 0x01, Foreign is TRUE | Default value |

| Site ADM element attribute | Attribute value computation | If not set, default value or error? |
|----------------------------|--|-------------------------------------|
| | Else if mSMQSiteForeign equals 0x00, Foreign is FALSE. | |
| MigratedFromMsmq10 | If mSMQnt4Stub ([MS-ADA2] section 2.528) equals 1, MigratedFromMsmq10 is TRUE Else if mSMQnt4Stub equals 0 MigratedFromMsmq10 is FALSE. | Error |
| Security | Security descriptor copied from nTSecurityDescriptor ([MS-ADA3] section 2.37). | Error |

3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.11.5 RoutingLink

1. A **RoutingLink** ADM element instance MUST be created, and *rDirectoryObject* MUST be set to this new **RoutingLink** ADM element instance.
2. The attributes listed in *iADMAttributeList* MUST be set on *rDirectoryObject*, using the attribute names in *iLDAPAttributeList* and the corresponding values in *iValues*, according to the following table. If the value of the LDAP attribute required to compute the **QueueManager** ADM element attribute is empty in *iValues*, depending on the attribute, either a default value MUST be supplied, or that is an error condition. <16> If an error condition occurs, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.

| RoutingLink ADM element attribute | Attribute value computation | If not set, default value or error? |
|-----------------------------------|--|-------------------------------------|
| Identifier | GUID copied from objectGUID ([MS-ADA3] section 2.44). | Error |
| Description | Unicode string transformed from description ([MS-ADA1] section 2.153). | Default value |
| FullPath | Distinguished name copied from distinguishedName ([MS-ADA1] section 2.177). | Error |
| ActualCost | Integer copied from mSMQCost ([MS-ADA2] section 2.507). | Error |
| Site1Identifier | Computed from mSMQSite1 ([MS-ADA2] section 2.547); see section 3.1.6.11.5.1. | Error |
| Site2Identifier | Computed from mSMQSite2 ([MS-ADA2] section 2.548); see section 3.1.6.11.5.2. | Error |
| SiteGateIdentifierList | Computed from mSMQSiteGates ([MS-ADA2] section 2.550); see section | Default value |

| RoutingLink ADM element attribute | Attribute value computation | If not set, default value or error? |
|-----------------------------------|-------------------------------|-------------------------------------|
| | 3.1.6.11.5.3. | |

3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.11.5.1 Site1Identifier

The value of the *rDirectoryObject.Site1Identifier* ADM attribute MUST be computed from the value of *mSMQSite1*:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "FullPath" EQUALS the value of *mSMQSite1*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **Identifier**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**:
 - *rStatus* must be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.
3. The value of the *rDirectoryObject.Site1Identifier* ADM attribute MUST be set to the value of the **Identifier** ADM attribute of the *rDirectoryObject* returned by the Read Directory event.

3.1.6.11.5.2 Site2Identifier

The value of the *rDirectoryObject.Site2Identifier* ADM attribute MUST be computed from the value of *mSMQSite2*:

1. Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - *iDirectoryObjectType* := "Site"
 - *iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "FullPath" EQUALS the value of *mSMQSite2*
 - *iAttributeList* := An array of the following **Site** ADM element attribute names:
 - **Identifier**
2. If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**:
 - *rStatus* must be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.

- The value of the *rDirectoryObject.Site2Identifier* ADM attribute MUST be set to the value of the **Identifier** ADM attribute of the *rDirectoryObject* returned by the Read Directory event.

3.1.6.11.5.3 SiteGateIdentifierList

The value of *rDirectoryObject.SiteGateIdentifierList* MUST be computed from the value of *mSMQSiteGates* according to the following algorithm:

- Let *TempList* be a list of distinguished names, initialized to the value of *rDirectoryObject.mSMQSiteGates*.
- Let *FinalList* be a list of GUIDs, initialized to be empty.
- For each distinguished name in *TempList*:
 - Generate a [Read Directory \(section 3.1.6.3\)](#) event with the following arguments:
 - iDirectoryObjectType* := "QueueManager"
 - iFilter* := An array of the following **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20):
 - "FullPath" EQUALS the current distinguished name from *TempList*
 - iAttributeList* := An array of the following **QueueManager** ADM element attribute names:
 - Identifier**
 - If the *rStatus* returned by the Read Directory event is not set to **DirectoryOperationResult.Success**:
 - rStatus* must be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.
 - The GUID in the **Identifier** ADM attribute of the *rDirectoryObject* returned by the Read Directory event MUST be placed in *FinalList*.
- The value of the *rDirectoryObject.SiteGateIdentifierList* attribute is the value of *FinalList*.

3.1.6.11.6 User

- A **User** ADM element instance MUST be created, and *rDirectoryObject* MUST be set to this new **User** ADM element instance.
- The attributes listed in *iADMAAttributeList* MUST be set on *rDirectoryObject*, using the attribute names in *iLDAPAttributeList* and the corresponding values in *iValues*, according to the following table. If the value of the LDAP attribute required to compute the **User** ADM element attribute is empty in *iValues*, depending on the attribute, either a default value MUST be supplied, or that is an error condition. [<17>](#) If an error condition occurs, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, *rDirectoryObject* is undefined, and processing MUST end.

| User ADM element attribute | Attribute value computation | If not set, default value or error? |
|----------------------------|---|-------------------------------------|
| Identifier | GUID copied from objectGUID ([MS-ADA3] section | Error |

| User ADM element attribute | Attribute value computation | If not set, default value or error? |
|------------------------------|---|-------------------------------------|
| | 2.44). | |
| SecurityIdentifier | SID copied from objectSid ([MS-ADA3] section 2.45). | Error |
| CertificateDigestList | List of GUIDs copied from mSMQDigests ([MS-ADA2] section 2.511). | Default value |
| Certificates | MQUSERSIGNCERTS ([MS-MQMQ] section 2.2.21) structure copied from mSMQSignCertificates ([MS-ADA2] section 2.544). | Default value |
| FullPath | Distinguished name copied from distinguishedName ([MS-ADA1] section 2.177). | Error |

3. *rStatus* MUST be set to **DirectoryOperationResult.Success**, and processing MUST end.

3.1.6.12 Create Object Using LDAP

This event MUST be generated with the following arguments:

- *iParentPath*: the distinguished name of the parent of the object to be created.
- *iChildName*: the name of the object to be created.
- *iObjectClass*: the **Active Directory schema** class of the new object.
- *iAttributes*: a list of name-value pairs.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.
- *rObjectGuid*: if *rStatus* is **DirectoryOperationResult.Success**, the value of the objectGUID attribute of the created object. Otherwise, this value is undefined and MUST NOT be used.

The algorithm MUST perform the following actions to process this event:

- If the value of the **CachedConfigurationNamingContext** ADM element is an empty string, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end.
- Raise a [Prepare an LDAP Connection \(section 3.1.6.18\)](#) event. The event takes no arguments. If the *rStatus* returned is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end. Otherwise, let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2), which is initialized to the value returned in *rADConnection*.
- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = searchRequest

- controls = none
- baseObject = *iParentPath*
- scope = baseObject
- derefAliases = neverDerefAliases
- sizeLimit = 0
- timeLimit = 0
- typesOnly = FALSE
- filter = "(objectClass=*)"
- attributes = a one-element list consisting of the string "objectClass"
- Perform the [Performing an LDAP Operation on an ADConnection](#) ([MS-ADSO] section 6.2.6.1.6) task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [RFC2251] section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section 2.2.6.
 - Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.
 - Processing MUST end.
- Let *ChildDistinguishedName* be a string variable, initialized by concatenating the strings "CN=", the value of *iChildName*, a comma ",", and the value of *iParentPath*.
- Construct a set of attribute name-value pairs:
 - The attribute objectClass MUST be paired with the value of *iObjectClass*.
 - All pairs in *iAttributes*.
- Construct an **LDAPMessage**:
 - messageID = set as described in [RFC2251] section 4.1.1.1.
 - protocolOp = addRequest
 - controls = none
 - entry = *ChildDistinguishedName*
 - attributes = the set of attribute name-value pairs constructed in the previous step
- Perform the Performing an LDAP Operation on an ADConnection task with the following parameters:

- *TaskInputADConnection* = *DirectoryServerConnection*
- *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- Construct an **LDAPMessage**:
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = searchRequest
 - controls = none
 - baseObject = *ChildDistinguishedName*
 - scope = baseObject
 - derefAliases = neverDerefAliases
 - sizeLimit = 0
 - timeLimit = 0
 - typesOnly = FALSE
 - filter = "(objectClass=*)"
 - attributes = an empty list
- Perform the Performing an LDAP Operation on an ADConnection task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- Extract the value of the objectGuid attribute from the result message returned in *TaskOutputResultMessages*, and set *rObjectGuid* to that value.
- Raise a Shut Down an LDAP Connection event.

- Processing MUST end.

3.1.6.13 Delete Object Using LDAP

This event MUST be generated with the following arguments:

- *iPath*: the distinguished name of the object to be deleted; optional if *iGuid* is supplied.
- *iGuid*: the value of the objectGuid attribute of the object to be deleted; optional if *iPath* is supplied.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.

The algorithm MUST perform the following actions to process this event:

- If the value of the **CachedConfigurationNamingContext** ADM element is an empty string, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end.
- Raise a [Prepare an LDAP Connection \(section 3.1.6.18\)](#) event. The event takes no arguments. If the *rStatus* returned is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end. Otherwise, let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2), which is initialized to the value returned in *rADConnection*.
- If *iGuid* is supplied, perform the following steps:
 - Raise a [Find Object By GUID Using LDAP \(section 3.1.6.19\)](#) event with the following arguments:
 - *iGuid* = *iGuid*
 - *iADConnection* = *DirectoryServerConnection*
 - If the value of *rStatus* returned by the event is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the value of *rStatus* returned by the event, and processing MUST end.
 - Set *iPath* to the value of *rDN*.
- Let *ParentPath* be a string variable that contains the distinguished name of the parent object of the object identified by *iPath* and that is initialized by copying the value of *iPath* and removing the leftmost comma-separated element.
- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = searchRequest
 - controls = none
 - baseObject = *ParentPath*
 - scope = baseObject
 - derefAliases = neverDerefAliases

- sizeLimit = 0
- timeLimit = 0
- typesOnly = FALSE
- filter = "(objectClass=*)"
- attributes = a one-element list consisting of the string "objectClass"
- Perform the [Performing an LDAP Operation on an ADConnection](#) ([MS-ADSO] section 6.2.6.1.6) task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [RFC2251](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.
 - Processing MUST end.
- Construct an **LDAPMessage**:
 - messageID = set as described in [RFC2251](#) section 4.1.1.1.
 - protocolOp = searchRequest
 - controls = none
 - baseObject = *iPath*
 - scope = baseObject
 - derefAliases = neverDerefAliases
 - sizeLimit = 0
 - timeLimit = 0
 - typesOnly = FALSE
 - filter = "(objectClass=*)"
 - attributes = a one-element list consisting of the string "objectClass"
- Perform the [Performing an LDAP Operation on an ADConnection](#) task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step

- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- Construct an **LDAPMessage**:
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = delRequest
 - controls = none
 - baseObject = *iPath*
- Perform the Performing an LDAP Operation on an ADConnection task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- Raise a Shut Down an LDAP Connection event.
- Processing MUST end.

3.1.6.14 Get Object Properties Using LDAP

This event MUST be generated with the following arguments:

- *iPath*: the distinguished name of the object; optional if *iGuid* is supplied.
- *iGuid*: the value of the objectGuid attribute of the object; optional if *iPath* is supplied.
- *iAttributes*: a list of attribute names.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.
- *rValues*: if *rStatus* is **Success**, the values of the attributes listed in *iAttributes*, in the same order. Otherwise, this value is undefined and MUST NOT be used.

The algorithm MUST perform the following actions to process this event:

- If the value of the **CachedConfigurationNamingContext** ADM element is an empty string, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end.
- If the *iAttributes* list does not contain the attribute name "objectGuid", add it to the list. If the *iAttributes* list does not contain the attribute name "distinguishedName", add it to the list.
- Raise a [Prepare an LDAP Connection \(section 3.1.6.18\)](#) event. The event takes no arguments. If the *rStatus* returned is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end. Otherwise, let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2), which is initialized to the value returned in *rADConnection*.
- If *iGuid* is supplied, perform the following steps:
 - Raise a [Find Object By GUID Using LDAP \(section 3.1.6.19\)](#) event with the following arguments:
 - *iGuid* = *iGuid*
 - *iADConnection* = *DirectoryServerConnection*
 - If the value of *rStatus* returned by the event is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the value of *rStatus* returned by the event, and processing MUST end.
 - Set *iPath* to the value of *rDN*.
- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - *messageID* = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - *protocolOp* = *searchRequest*
 - *controls* = none
 - *baseObject* = *iPath*
 - *scope* = *baseObject*
 - *derefAliases* = *neverDerefAliases*
 - *sizeLimit* = 0
 - *timeLimit* = 0
 - *typesOnly* = FALSE
 - *filter* = "(objectClass=*)"
 - *attributes* = an empty list
- Perform the [Performing an LDAP Operation on an ADConnection](#) ([\[MS-ADSO\]](#) section 6.2.6.1.6) task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step

- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.
 - Processing MUST end.
- For each attribute name in *iAttributes*, extract the value for that attribute from the result message returned in *TaskOutputResultMessages* and add it to the *rValues* list. If there is no matching value in the results, add an empty entry to the *rValues* list.
- Raise a Shut Down an LDAP Connection event.
- Processing MUST end.

3.1.6.15 Search Using LDAP

This event MUST be generated with the following arguments:

- *iObjectClass*: the Active Directory schema class of the objects being searched for.
- *iFilter*: An array of **attribute-filter expressions** ([\[MS-MQDMPR\]](#) section 3.1.7.1.20), but modified so that the valid attributes comprise the set of Active Directory attributes associated with an Active Directory schema class instance of type *iObjectClass* instead of ADM element attributes. Each sublist that is returned in *rValues* MUST satisfy all attribute-filter expressions in this array.
- *iAttributes*: a list of attribute names.

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.
- *rValues*: if *rStatus* is **Success**, a list of sublists, where each sublist corresponds to one object found in Active Directory and is composed of values for that object of the attributes listed in *iAttributes*, in the same order. Otherwise, this value is undefined and MUST NOT be used.

The algorithm MUST perform the following actions to process this event:

- If the value of the **CachedConfigurationNamingContext** ADM element is an empty string, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end.
- If the *iAttributes* list does not contain the attribute name "objectGuid", add it to the list. If the *iAttributes* list does not contain the attribute name "distinguishedName", add it to the list.
- Raise a [Prepare an LDAP Connection \(section 3.1.6.18\)](#) event. The event takes no arguments. If the *rStatus* returned is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end. Otherwise, let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2), which is initialized to the value returned in *rADConnection*.
- Let *BaseDN* be a distinguished name composed depending on the class name in *iObjectClass* according to these rules:

- If *iObjectClass* is "mSMQConfiguration", "mSMQQueue", or "user", *BaseDN* MUST be set to the value of the **CachedConfigurationNamingContext** ADM element.
- If *iObjectClass* is "mSMQEnterpriseSettings", *BaseDN* MUST be set to a string formed by concatenating "CN=Services," and the value of the **CachedConfigurationNamingContext** ADM element.
- If *iObjectClass* is "site", *BaseDN* MUST be set to a string formed by concatenating "CN=Sites," and the value of the **CachedConfigurationNamingContext** ADM element.
- If *iObjectClass* is "mSMQSiteLink", *BaseDN* MUST be set to a string formed by concatenating "CN=MsmqServices,CN=Services," and the value of the **CachedConfigurationNamingContext** ADM element.
- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = searchRequest
 - controls = none
 - baseObject = *BaseDN*
 - scope = baseObject
 - derefAliases = neverDerefAliases
 - sizeLimit = 0
 - timeLimit = 0
 - typesOnly = FALSE
 - filter = "(objectClass=*)"
 - attributes = a one-element list consisting of the string "objectClass"
- Perform the [Performing an LDAP Operation on an ADConnection](#) ([\[MS-ADSO\]](#) section 6.2.6.1.6) task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.
 - Processing MUST end.
- Construct an **LDAPMessage**:
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.

- `protocolOp = searchRequest`
- `controls = none`
- `baseObject = BaseDN`
- `scope = wholeSubtree`
- `derefAliases = neverDerefAliases`
- `sizeLimit = 0`
- `timeLimit = 120`
- `typesOnly = FALSE`
- `filter = iFilter`
- `attributes = iAttributes`
- Perform the Performing an LDAP Operation on an ADConnection task with the following parameters:
 - `TaskInputADConnection = DirectoryServerConnection`
 - `TaskInputRequestMessage` = the **LDAPMessage** constructed in the preceding step
- If the value of `TaskReturnStatus` is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - `rStatus` MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for `TaskReturnStatus` specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- `TaskOutputResultMessages` contains a list of attributes for each object in the directory matching the search filter. For each such list returned, perform the following steps:
 - Let `Sublist` be a list of values, initialized to be empty.
 - For each attribute name in `iAttributes`, find the value for that attribute in the results list and add it to `Sublist`. If there is no matching value in the results, add an empty entry to `Sublist`.
 - Add `Sublist` to `rValues`.
- Raise a Shut Down an LDAP Connection event.
- Processing MUST end.

3.1.6.16 Set Object Properties Using LDAP

This event MUST be generated with the following arguments:

- `iPath`: the distinguished name of the object; optional if `iGuid` is supplied
- `iGuid`: the value of the objectGuid attribute of the object; optional if `iPath` is supplied

- *iAttributes*: a list of name-value pairs

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.

The algorithm MUST perform the following actions to process this event:

- If the value of the **CachedConfigurationNamingContext** ADM element is an empty string, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end.
- Raise a [Prepare an LDAP Connection \(section 3.1.6.18\)](#) event. The event takes no arguments. If the *rStatus* returned is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end. Otherwise, let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2), which is initialized to the value returned in *rADConnection*.
- If *iGuid* is supplied, perform the following steps:
 - Raise a [Find Object By GUID Using LDAP \(section 3.1.6.19\)](#) event with the following arguments:
 - *iGuid* = *iGuid*
 - *iADConnection* = *DirectoryServerConnection*
 - If the value of *rStatus* returned by the event is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the value of *rStatus* returned by the event, and processing MUST end.
 - Set *iPath* to the value of *rDN*.
- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - *messageID* = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - *protocolOp* = *searchRequest*
 - *controls* = none
 - *baseObject* = *iPath*
 - *derefAliases* = *neverDerefAliases*
 - *sizeLimit* = 0
 - *timeLimit* = 0
 - *scope* = *baseObject*
 - *typesOnly* = FALSE
 - *filter* = "(objectClass=*)"
 - *attributes* = a one-element list consisting of the string "objectClass"
- Perform the [Performing an LDAP Operation on an ADConnection](#) ([\[MS-ADSO\]](#) section 6.2.6.1.6) task with the following parameters:

- *TaskInputADConnection* = *DirectoryServerConnection*
- *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.
 - Processing MUST end.
- Construct an **LDAPMessage**:
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = modifyRequest
 - controls = none
 - baseObject = *iPath*
 - operation = replace
 - modification = *iAttributes*
- Perform the Performing an LDAP Operation on an ADConnection task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- Raise a Shut Down an LDAP Connection event.
- Processing MUST end.

3.1.6.17 Set Object Security Using LDAP

This event MUST be generated with the following arguments:

- *iPath*: the distinguished name of the object; optional if *iGuid* is supplied.
- *iGuid*: the value of the objectGuid attribute of the object; optional if *iPath* is supplied.
- *iSecurity*: a **SECURITY_DESCRIPTOR** ([\[MS-DTYP\]](#) section 2.4.6).

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.

The algorithm MUST perform the following actions to process this event:

- If the value of the **CachedConfigurationNamingContext** ADM element is an empty string, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end.
- Raise a [Prepare an LDAP Connection \(section 3.1.6.18\)](#) event. The event takes no arguments. If the *rStatus* returned is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to **DirectoryOperationResult.DirectoryNotConnected**, and processing MUST end. Otherwise, let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2), which is initialized to the value returned in *rADConnection*.
- If *iGuid* is supplied, perform the following steps:
 - Raise a [Find Object By GUID Using LDAP \(section 3.1.6.19\)](#) event with the following arguments:
 - *iGuid* = *iGuid*
 - *iADConnection* = *DirectoryServerConnection*
 - If the value of *rStatus* returned by the event is not **DirectoryOperationResult.Success**, *rStatus* MUST be set to the value of *rStatus* returned by the event, and processing MUST end.
 - Set *iPath* to the value of *rDN*.
- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - *messageID* = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - *protocolOp* = *searchRequest*
 - *controls* = *none*
 - *baseObject* = *iPath*
 - *scope* = *baseObject*
 - *derefAliases* = *neverDerefAliases*
 - *sizeLimit* = 0
 - *timeLimit* = 0
 - *typesOnly* = *FALSE*
 - *filter* = "(objectClass=*)"
 - *attributes* = a one-element list consisting of the string "objectClass"
- Perform the [Performing an LDAP Operation on an ADConnection](#) ([\[MS-ADSO\]](#) section 6.2.6.1.6) task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*

- *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.
 - Processing MUST end.
- Construct an **LDAPMessage**:
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = searchRequest
 - controls = none
 - baseObject = *iPath*
 - scope = baseObject
 - derefAliases = neverDerefAliases
 - sizeLimit = 0
 - timeLimit = 0
 - typesOnly = FALSE
 - filter = "(nTSecurityDescriptor=*)"
 - attributes = a one-element list consisting of the string "objectClass"
- Perform the Performing an LDAP Operation on an ADConnection task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- Construct an **LDAPMessage**:
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = modifyRequest

- controls = none
- baseObject = *iPath*
- operation = replace
- modification = an attribute-value pair consisting of "ntSecurityDescriptor" and *iSecurity*
- Perform the Performing an LDAP Operation on an ADConnection task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a Shut Down an LDAP Connection event.
 - Processing MUST end.
- Raise a Shut Down an LDAP Connection event.
- Processing MUST end.

3.1.6.18 Prepare an LDAP Connection

This event MUST be generated with no arguments.

Return Values:

- *rADConnection*: An **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2) that is connected to an Active Directory server and that is ready to perform LDAP operations. The value is undefined if *rStatus* is not **DirectoryOperationResult.Success**.
- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation.

The algorithm MUST perform the following actions to process this event:

- If the **CachedLDAPConnection.Handle** (section [3.1.1.6](#)) ADM element attribute is not NULL:
 - Increment the value of the **CachedLDAPConnection.RefCount** ADM element attribute by 1.
 - Set *rADConnection* to the value of **CachedLDAPConnection.Handle**, and set *rStatus* to **DirectoryOperationResult.Success**.
 - Take no further action.
- Let *DirectoryServerConnection* be a variable of type **ADCONNECTION_HANDLE**.
- Perform the [Initializing an ADConnection](#) ([\[MS-ADSO\]](#) section 6.2.6.1.1) task, specifying the following parameters:
 - *TaskInputTargetName* = NULL

- *TaskInputPortNumber* = 389
- *DirectoryServerConnection* is set to the *TaskReturnADConnection* result returned by the task.
- Perform the [Setting an LDAP Option on an ADConnection](#) ([MS-ADSO] section 6.2.6.1.2) task, specifying the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputOptionName* = "LDAP_OPT_PROTOCOL_VERSION"
 - *TaskInputOptionValue* = 3
- Perform the [Establishing an ADConnection](#) ([MS-ADSO] section 6.2.6.1.3) task, specifying the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
- If the *TaskReturnStatus* result is not success, as specified in [RFC2251](#) section 4.1.10, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
- Perform the [Performing an LDAP Bind on an ADConnection](#) ([MS-ADSO] section 6.2.6.1.4) task, specifying the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
- If the *TaskReturnStatus* result is not success, as specified in [RFC2251](#) section 4.1.10, *rStatus* MUST be set to **DirectoryOperationResult.GenericError**, and processing MUST end.
- *rADConnection* MUST be set to *DirectoryServerConnection*, and *rStatus* MUST be set to **DirectoryOperationResult.Success**.
- Set the **CachedLDAPConnection.Handle** ADM element attribute to the value of *rADConnection*, and set the **CachedLDAPConnection.RefCount** ADM element attribute to 1.

3.1.6.19 Find Object By GUID Using LDAP

This event MUST be generated with the following arguments:

- *iGuid*: the value of the objectGuid attribute of the requested object.
- *iADConnection*: An **ADCONNECTION_HANDLE** ([MS-DTYP] section 2.2.2).

Return Values:

- *rStatus*: A **DirectoryOperationResult** that indicates the result of this directory operation. If this value is not **DirectoryOperationResult.Success**, the **ADConnection** in *iADConnection* is no longer bound and MUST NOT be used to perform further operations, and the value of *rDN* is undefined.
- *rDN*: The distinguished name of the object found.

The algorithm MUST perform the following actions to process this event:

- Let *CompactGuid* be a string variable that is initialized by converting the value of *iGuid* to curly braced GUID string form ([MS-DTYP] section 2.3.4.3), then compacting it by removing all characters that are not hexadecimal digits.

- Construct an **LDAPMessage** ([\[RFC2251\]](#) section 4.1):
 - messageID = set as described in [\[RFC2251\]](#) section 4.1.1.1.
 - protocolOp = searchRequest
 - controls = none
 - baseObject = an empty string
 - scope = wholeSubtree
 - derefAliases = neverDerefAliases
 - sizeLimit = 0
 - timeLimit = 0
 - typesOnly = FALSE
 - filter = "(objectGuid=" *CompactGuid* ")"
 - attributes = a one-element list consisting of the string "distinguishedName"
- Perform the [Performing an LDAP Operation on an ADConnection](#) ([\[MS-ADSO\]](#) section 6.2.6.1.6) task with the following parameters:
 - *TaskInputADConnection* = *DirectoryServerConnection*
 - *TaskInputRequestMessage* = the **LDAPMessage** constructed in the preceding step
- If the value of *TaskReturnStatus* is not success, as defined in [\[RFC2251\]](#) section 4.1.10, the algorithm MUST perform the following steps:
 - *rStatus* MUST be set to a **DirectoryOperationResult** enumeration value according to the conversion rules for *TaskReturnStatus* specified in section [2.2.6](#).
 - Raise a [Shut Down an LDAP Connection \(section 3.1.6.21\)](#) event.
 - Processing MUST end.
- Extract the value of the distinguishedName attribute from the result message returned in *TaskOutputResultMessages* and set *rDN* to that value.
- Set *rStatus* to **DirectoryOperationResult.Success**.
- Processing MUST end.

3.1.6.20 Data Element Directory Attribute Tables

3.1.6.20.1 QueueManager

The **QueueManager** ADM element is specified in [\[MS-MQDMPR\]](#) section 3.1.1.1. The following table lists only those **QueueManager** ADM element attributes that are stored in the directory.

QueueManager ADM element attributes MUST be stored as attributes of an mSMQConfiguration ([\[MS-ADSC\]](#) section 2.157) object, except for those noted in the following table, which MUST be stored on the computer ([\[MS-ADSC\]](#) section 2.21) object that is the parent of the

mSMQConfiguration object in the directory. In addition, any **QueueManager** ADM element instance for which one or more of the **RoutingServer**, **DirectoryServer**, or **SupportingServer** ADM attributes is TRUE MUST have an mSMQSettings ([\[MS-ADSC\]](#) section 2.161) object in the directory, the attributes of which MUST have the same values as certain attributes of the mSMQConfiguration object. The distinguished names for these three objects (mSMQConfiguration, computer, and mSMQSettings) are specified in section [2.2.1](#).

| QueueManager ADM element attribute | Stored on computer object? |
|---------------------------------------|--|
| Identifier | no |
| ComputerName | no |
| QueueManagerVersion | no |
| OperatingSystemType | no |
| QualifiedComputerName | yes |
| CreateTime | no |
| ModifyTime | no |
| DirectoryServerType | no |
| RemoteAccessServer | no |
| QueueManagerQuota | no |
| JournalQuota | no |
| ForeignSystem | no |
| FullPath | no |
| SiteIdentifierList | no |
| ConnectedNetworkIdentifierList | This ADM element attribute is not supported in Active Directory-based environments. Any attempt to operate on it results in DirectoryOperationResult.GenericError . |
| OutRoutingServerIdentifierList | no |
| InRoutingServerIdentifierList | no |
| RoutingServer | no |
| DirectoryServer | no |
| SupportingServer | no |
| PublicEncryptionKeyList | no |
| PublicSigningKeyList | yes |
| Security | no |
| Clustered | yes |

| QueueManager ADM element attribute | Stored on computer object? |
|------------------------------------|----------------------------|
| OperatingSystemVersion | yes |

3.1.6.20.2 Queue

The **Queue** ADM element is specified in [\[MS-MQDMPR\]](#) section 3.1.1.2. The following table lists only those **Queue** ADM element attributes that are stored in the directory.

Queue ADM elements MUST be stored in the directory if the value of the **QueueType** ADM attribute is **Public**. **Queue** ADM elements MUST NOT be stored in the directory if the value of the **QueueType** ADM attribute is not **Public**.

| Queue ADM element attribute |
|-----------------------------|
| Identifier |
| Label |
| CreateTime |
| ModifyTime |
| Type |
| Pathname |
| QualifiedPathname |
| Journaling |
| Quota |
| JournalQuota |
| Authentication |
| PrivacyLevel |
| Transactional |
| MulticastAddress |
| Security |
| BasePriority |
| FullPath |
| DirectoryPath |

3.1.6.20.3 Enterprise

The **Enterprise** ADM element is specified in [\[MS-MQDMPR\]](#) section 3.1.1.6. The following table lists only those **Enterprise** ADM element attributes that are stored in the directory.

| Enterprise ADM element attribute |
|--|
| Identifier |
| Name |
| WeakenedSecurity |
| NonLDAPCapableQueueManagerNotification |
| DefaultTimeToLive |
| OldDirectory |
| Security |

3.1.6.20.4 Site

The **Site** ADM element is specified in [\[MS-MQDMPR\]](#) section 3.1.1.7. The following table lists only those **Site** ADM element attributes that are stored in the directory.

| Site ADM element attribute | Notes |
|------------------------------|--|
| Identifier | |
| Name | |
| PrimarySiteController | This ADM element attribute is not supported in Active Directory-based environments. Any attempt to operate on it results in DirectoryOperationResult.GenericError . |
| IntraSiteReplicationInterval | |
| InterSiteReplicationInterval | |
| FullPath | |
| ForeignSite | |
| PublicSigningKeyList | This ADM element attribute is not supported in Active Directory-based environments. Any attempt to operate on it results in DirectoryOperationResult.GenericError . |
| MigratedFromMsmq10 | |
| Security | |

3.1.6.20.5 RoutingLink

The **RoutingLink** ADM element is specified in [\[MS-MQDMPR\]](#) section 3.1.1.8. The following table lists only those **RoutingLink** ADM attributes that are stored in the directory.

| RoutingLink ADM element attribute |
|-----------------------------------|
| Identifier |
| Description |

| RoutingLink ADM element attribute |
|-----------------------------------|
| FullPath |
| ActualCost |
| Site1Identifier |
| Site2Identifier |
| SiteGateIdentifierList |

3.1.6.20.6 User

The **User** ADM element is specified in [\[MS-MQDMPR\]](#) section 3.1.1.15 and extended in [User Data Element \(section 3.1.1.4\)](#). The following table lists only those **User** ADM element attributes that are stored in the directory.

| User ADM element attribute |
|----------------------------|
| Identifier |
| SecurityIdentifier |
| CertificateDigestList |
| Certificates |
| FullPath |

3.1.6.21 Shut Down an LDAP Connection

This event MUST be generated with no arguments.

Return Values:

- None.

The algorithm MUST perform the following actions to process this event:

- Decrement the value of the **CachedLDAPConnection.RefCount** (section [3.1.1.6](#)) ADM element attribute by 1.
- If the value of the **CachedLDAPConnection.RefCount** ADM element attribute is greater than zero, take no further action.
- Let *DirectoryServerConnection* be an **ADCONNECTION_HANDLE** ([\[MS-DTYP\]](#) section 2.2.2) that is initialized to the value of the **CachedLDAPConnection.Handle** ADM element attribute.
- Set the **CachedLDAPConnection.Handle** ADM element attribute to NULL.
- Perform the [Performing an LDAP Unbind on an ADConnection](#) ([\[MS-ADSO\]](#) section 6.2.6.1.5) task with the following parameter:
 - *TaskInputADConnection* = *DirectoryServerConnection*

4 Algorithm Examples

None.

5 Security

5.1 Security Considerations for Implementers

Active Directory requires authentication to access the directory via LDAP. This authentication is performed via SASL, using the GSS-SPNEGO protocol as described in [\[MS-ADTS\]](#) section 5.1.1.

Active Directory performs authorization on each access to each object in the directory, as described in [\[MS-ADTS\]](#) section 5.1.3. The **discretionary access control list (DACL)**, if any, found in the `nTSecurityDescriptor` attribute ([\[MS-ADA3\]](#) section 2.37) of the object is used in this process. This algorithm controls that DACL only for the `mSMQConfiguration` ([\[MS-ADSC\]](#) section 2.157), `mSMQQueue` ([\[MS-ADSC\]](#) section 2.160), `site` ([\[MS-ADSC\]](#) section 2.252), and `mSMQEnterpriseSettings` ([\[MS-ADSC\]](#) section 2.158) objects; for all of the other Active Directory objects listed in section [2.2.1](#), the defaults supplied by Active Directory are used.

The following sections describe, for each combination of object and operation, the requested access mask that is compared to the granted rights in the DACL evaluation process used by Active Directory and described in [\[MS-ADTS\]](#) section 5.1.3.3.2. If the required access is not granted, Active Directory returns a failure.

5.1.1 QueueManager

The directory representation of a **QueueManager** ([\[MS-MQDMPR\]](#) section 3.1.1) ADM element instance can be created, deleted, read, and modified, as described in sections [3.1.6.1.1](#), [3.1.6.2.1](#), [3.1.6.3.2](#), [3.1.6.4.1](#), and [3.1.6.7.1](#). These operations always act on an `mSMQConfiguration` ([\[MS-ADSC\]](#) section 2.157) Active Directory object and can also act on `computer` ([\[MS-ADSC\]](#) section 2.21) and `mSMQSettings` ([\[MS-ADSC\]](#) section 2.161) objects. The default security is discussed in section [3.1.6.1.1.8](#).

To create the directory representation of a **QueueManager** ADM element instance requires at least `RIGHT_DS_CREATE_CHILD` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the parent `computer` object. If the **PublicSigningKeyList** ADM attribute is present, as described in section [3.1.6.1.1.3](#), `RIGHT_GENERIC_WRITE` ([\[MS-ADTS\]](#) section 5.1.3.2) access is also required on the parent object. If one or more associated `mSMQSettings` objects are required to be created, as described in section [3.1.6.1.1.3](#), `RIGHT_DS_CREATE_CHILD` access is required on the parent objects of the `mSMQSettings` objects, which are of class `server` ([\[MS-ADSC\]](#) [2.244](#)).

To delete the directory representation of a **QueueManager** ADM element instance requires `RIGHT_DS_CREATE_CHILD` access on the parent object and `RIGHT_DELETE` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the `mSMQConfiguration` object itself. If one or more associated `mSMQSettings` objects exist, as described in section [3.1.6.2.1.3](#), `RIGHT_DS_DELETE_CHILD` ([\[MS-ADTS\]](#) section 5.1.3.2) access is required on the parent objects of the `mSMQSettings` objects, and `RIGHT_DELETE` access on the `mSMQSettings` objects themselves.

To read the attributes of a **QueueManager** ADM element instance requires `RIGHT_GENERIC_READ` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object. If any of the **QualifiedComputerName**, **PublicSigningKeyList**, **OperationSystemVersion**, or **Clustered** ADM attributes is specified, `RIGHT_GENERIC_READ` access on the parent object is also required.

To modify the directory representation of a **QueueManager** ADM element instance requires `RIGHT_GENERIC_WRITE` access on the object. If the **PublicSigningKeyList** ADM attribute is to be modified, `RIGHT_GENERIC_WRITE` access on the parent object is also required. As a result of this modification, one or more associated `mSMQSettings` objects can also be created, modified, or deleted, as described in section [3.1.6.7.1.9](#). To create `mSMQSettings` objects, `RIGHT_DS_CREATE_CHILD` access is required on the parent objects of the `mSMQSettings` objects.

To modify, `RIGHT_GENERIC_WRITE` access is required on the `mSMQSettings` objects. To delete, `RIGHT_DS_DELETE_CHILD` access is required on the parent objects of the `mSMQSettings` objects, and `RIGHT_DELETE` access on the `mSMQSettings` objects themselves.

5.1.2 Queue

The directory representation of a **Queue** ([\[MS-MQDMPR\]](#) section 3.1.1.2) ADM element instance can be created, deleted, read, and modified, as described in sections [3.1.6.1.2](#), [3.1.6.2.2](#), [3.1.6.3.3](#), [3.1.6.4.2](#), and [3.1.6.7.2](#). These operations act on an `mSMQQueue` ([\[MS-ADSC\]](#) section 2.160) Active Directory object. The default security is discussed in [\[MS-MQDMPR\]](#) section 3.1.7.1.3.1.

To create the directory representation of a **Queue** ADM element instance requires `RIGHT_DS_CREATE_CHILD` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the parent `mSMQConfiguration` ([\[MS-ADSC\]](#) section 2.157) object.

To delete the directory representation of a **Queue** ADM element instance requires `RIGHT_DS_DELETE_CHILD` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the parent object and `RIGHT_DELETE` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the `mSMQQueue` object itself.

To read the attributes of a **Queue** ADM element instance requires `RIGHT_GENERIC_READ` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

To modify the directory representation of a **Queue** ADM element instance requires `RIGHT_GENERIC_WRITE` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

5.1.3 Enterprise

The directory representation of an **Enterprise** ([\[MS-MQDMPR\]](#) section 3.1.1.6) ADM element instance can be read and modified, as described in sections [3.1.6.3.4](#), [3.1.6.4.3](#), and [3.1.6.7.3](#). These operations act on an `mSMQEnterpriseSettings` ([\[MS-ADSC\]](#) section 2.158) Active Directory object.

To read the attributes of an **Enterprise** ADM element instance requires `RIGHT_GENERIC_READ` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

To modify the directory representation of an **Enterprise** ADM element instance requires `RIGHT_GENERIC_WRITE` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

5.1.4 Site

The directory representation of a **Site** ([\[MS-MQDMPR\]](#) section 3.1.1.7) ADM element instance can be created, deleted, read, and modified, as described in sections [3.1.6.1.3](#), [3.1.6.2.3](#), [3.1.6.3.5](#), [3.1.6.4.4](#), and [3.1.6.7.4](#). These operations act on a site ([\[MS-ADSC\]](#) section 2.252) Active Directory object.

To create the directory representation of a **Site** ADM element instance requires `RIGHT_DS_CREATE_CHILD` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the parent object, which is of class `sitesContainer` ([\[MS-ADSC\]](#) section 2.255) and has a fixed distinguished name of "CN=Sites,CN=Configuration,<root>".

To delete the directory representation of a **Site** ADM element instance requires `RIGHT_DS_DELETE_CHILD` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the parent object and `RIGHT_DELETE` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the site object itself.

To read the attributes of a **Site** ADM element instance requires `RIGHT_GENERIC_READ` ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

To modify the directory representation of a **Site** ADM element instance requires RIGHT_GENERIC_WRITE ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

5.1.5 RoutingLink

The directory representation of a **RoutingLink** ([\[MS-MQDMPR\]](#) section 3.1.1.8) ADM element instance can be created, deleted, read, and modified, as described in sections [3.1.6.1.4](#), [3.1.6.2.4](#), [3.1.6.3.6](#), [3.1.6.4.5](#), and [3.1.6.7.5](#). These operations act on an mSMQSiteLink ([\[MS-ADSC\]](#) section 2.162) Active Directory object.

To create the directory representation of a **RoutingLink** ADM element instance requires RIGHT_DS_CREATE_CHILD ([\[MS-ADTS\]](#) section 5.1.3.2) access on the parent mSMQEnterpriseSettings ([\[MS-ADSC\]](#) section 2.158) object.

To delete the directory representation of a **RoutingLink** ADM element instance requires RIGHT_DS_DELETE_CHILD ([\[MS-ADTS\]](#) section 5.1.3.2) access on the mSMQEnterpriseSettings object and RIGHT_DELETE ([\[MS-ADTS\]](#) section 5.1.3.2) access on the mSMQSiteLink object itself.

To read the attributes of a **RoutingLink** ADM element instance requires RIGHT_GENERIC_READ ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

To modify the directory representation of a **RoutingLink** ADM element instance requires RIGHT_GENERIC_WRITE ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

5.1.6 User

The directory representation of a **User** ([\[MS-MQDMPR\]](#) section 3.1.1.15) ADM element instance can be read and modified, as described in sections [3.1.6.3.7](#), [3.1.6.4.6](#), and [3.1.6.7.6](#). These operations act on a user ([\[MS-ADSC\]](#) section 2.263) Active Directory object.

To read the attributes of a **User** ADM element instance requires RIGHT_GENERIC_READ ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

To modify the directory representation of a **User** ADM element instance requires RIGHT_GENERIC_WRITE ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

5.1.7 Queue Alias

A [queue alias \(section 2.3\)](#) can be read, as described in section [3.1.6.8](#). This operation acts on an mSMQ-Custom-Recipient Active Directory object and requires RIGHT_GENERIC_READ ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

5.1.8 Distribution List

A distribution list can be read, as described in section [3.1.6.9](#). This operation acts on a group ([\[MS-ADSC\]](#) section 2.53) object and requires RIGHT_GENERIC_READ ([\[MS-ADTS\]](#) section 5.1.3.2) access on the object.

5.2 Index of Security Parameters

None.

6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Windows 2000 operating system
- Windows XP operating system
- Windows Server 2003 operating system
- Windows Server 2003 R2 operating system
- Windows Vista operating system
- Windows Server 2008 operating system
- Windows 7 operating system
- Windows Server 2008 R2 operating system
- Windows 8 operating system
- Windows Server 2012 operating system
- Windows 8.1 operating system
- Windows Server 2012 R2 operating system

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

[<1> Section 2.2.2:](#) The directory attributes mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.511), mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545), mSMQSiteGatesMig ([\[MS-ADA2\]](#) section 2.551), and mSMQMigrated ([\[MS-ADA2\]](#) section 2.525) are used only on Windows 2000.

[<2> Section 2.2.2:](#) The directory attributes mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.511), mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545), mSMQSiteGatesMig ([\[MS-ADA2\]](#) section 2.551), and mSMQMigrated ([\[MS-ADA2\]](#) section 2.525) are used only on Windows 2000.

[<3> Section 2.2.2:](#) The directory attributes mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.511), mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545), mSMQSiteGatesMig ([\[MS-ADA2\]](#) section 2.551), and mSMQMigrated ([\[MS-ADA2\]](#) section 2.525) are used only on Windows 2000.

[<4> Section 2.2.2:](#) The directory attributes mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.511), mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545), mSMQSiteGatesMig ([\[MS-ADA2\]](#) section 2.551), and mSMQMigrated ([\[MS-ADA2\]](#) section 2.525) are used only on Windows 2000.

[<5> Section 2.2.2:](#) The directory attributes mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.511), mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545), mSMQSiteGatesMig ([\[MS-ADA2\]](#) section 2.551), and mSMQMigrated ([\[MS-ADA2\]](#) section 2.525) are used only on Windows 2000.

<6> [Section 2.2.2](#): The directory attributes mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.511), mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545), mSMQSiteGatesMig ([\[MS-ADA2\]](#) section 2.551), and mSMQMigrated ([\[MS-ADA2\]](#) section 2.525) are used only on Windows 2000.

<7> [Section 2.2.3](#): For Windows 2000, Windows Server 2003, Windows Server 2008, and Windows Server 2008 R2, an mSMQEnterpriseSettings object is created as part of the creation of each new rootDomainNamingContext.

<8> [Section 2.3](#): Windows NT, Windows 2000, and Windows XP do not use the queue alias object. Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 use the Windows API function **MQADsPathToFormatName** to translate a directory service path of a queue alias into its associated format name.

<9> [Section 3.1.6.1.1.9](#): The directory attributes mSMQSignCertificatesMig ([\[MS-ADA2\]](#) section 2.545) and mSMQDigestsMig ([\[MS-ADA2\]](#) section 2.512) are set only on Windows 2000.

<10> [Section 3.1.6.11.1](#): Default values are used by Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2. The default value of the **QueueManagerQuota** ADM element for Windows XP is 0xFFFFFFFF.

<11> [Section 3.1.6.11.1](#): Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 use an empty Unicode string as the default value.

<12> [Section 3.1.6.11.1](#): The default value used by Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 is an empty list.

<13> [Section 3.1.6.11.3](#): This is a table of the default values used by Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2.

| Enterprise ADM element attribute | Default value |
|---|---|
| WeakenedSecurity | See section 3.1.6.11.3.2 to convert a mSMQNameStyle value of 0x02. |
| NonLDAPCapableQueueManagerNotification | "Y" |
| DefaultTimeToLive | <ul style="list-style-type: none">▪ 7776000 (Windows XP)▪ 345600 (Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2) |
| OldDirectory | TRUE |

<14> [Section 3.1.6.11.3.2](#): Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 use an internal default that is equivalent to an mSMQNameStyle value of 0x00.

<15> [Section 3.1.6.11.4](#): This is a table of the default values used by Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2.

| Site ADM element attribute | Default value |
|-------------------------------------|---------------|
| IntraSiteReplicationInterval | 2 |
| InterSiteReplicationInterval | 10 |
| Foreign | FALSE |

<16> [Section 3.1.6.11.5](#): This is a table of the default values used by Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2.

| mSMQSiteLink ADM element attribute | Default value |
|------------------------------------|---------------|
| Description | empty string |
| SiteGateIdentifierList | empty list |

<17> [Section 3.1.6.11.6](#): This is a table of the default values used by Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2.

| User ADM element attribute | Default value |
|------------------------------|---------------|
| Certificates | empty |
| CertificateDigestList | empty list |

7 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

8 Index

A

Abstract data model
data elements
[CachedConfigurationNamingContext](#) 29
[CachedLDAPConnection](#) 29
[externally defined](#) 28
[ReadDirectoryIteratorState](#) 28
[ReadDirectoryIteratorStateCollection](#) 29
[user](#) 29
[overview](#) 28
[Algorithm - overview](#) 28
[Applicability](#) 11
[Attributes of directory objects](#) 13

C

[Capability negotiation](#) 11
[Change tracking](#) 140
Common data types
directory objects
[attributes](#) 13
[LDAP distinguished names](#) 12
[hash string calculation from queue name](#) 15
[mSMQEnterpriseSettings object](#) 15
[unused active directory properties](#) 15

D

Data model - abstract
data elements
[CachedConfigurationNamingContext](#) 29
[CachedLDAPConnection](#) 29
[externally defined](#) 28
[ReadDirectoryIteratorState](#) 28
[ReadDirectoryIteratorStateCollection](#) 29
[user](#) 29
[overview](#) 28
Data types - common
directory objects
[attributes](#) 13
[LDAP distinguished names](#) 12
[hash string calculation from queue name](#) 15
[mSMQEnterpriseSettings object](#) 15
[unused active directory properties](#) 15
Directory
objects
[attributes](#) 13
[LDAP distinguished names](#) 12
[properties - unused - active](#) 15
[service schema elements](#) 25

E

[Elements - directory service schema](#) 25
[Examples](#) 133

F

[Fields - vendor-extensible](#) 11

G

[Glossary](#) 9

H

[Hash string calculation from queue name](#) 15

I

[Implementer - security considerations](#) 134
[Index of security parameters](#) 136
[Informative references](#) 10
[Initialization](#) 29
[Introduction](#) 9

L

[LDAP distinguished names of directory objects](#) 12
Local events
create
[ADM element From LDAP values](#) 97
[directory object](#) 31
[LDAP attribute list](#) 93
[object using LDAP](#) 112
delete
[directory object](#) 46
[object using LDAP](#) 115
[find object by GUID using LDAP](#) 127
[get object properties using LDAP](#) 117
LDAP connection
[prepare](#) 126
[shut down](#) 132
read
[directory](#) 50
[directory begin](#) 57
[directory end](#) 72
[directory next](#) 71
resolve
[distribution list](#) 92
[queue alias](#) 91
[search using LDAP](#) 119
set object
[properties using LDAP](#) 121
[security using LDAP](#) 123
[write directory](#) 72

M

[Message processing](#) 30
Messages
common data types
directory objects
[attributes](#) 13
[LDAP distinguished names](#) 12
[hash string calculation from queue name](#) 15
[mSMQEnterpriseSettings object](#) 15

[unused active directory properties](#) 15
[queue alias](#) 25
[transport](#) 12
[mSMQEnterpriseSettings object](#) 15

N

[Normative references](#) 10

O

[Objects - mSMQEnterpriseSettings](#) 15
[Overview \(synopsis\)](#) 10

P

[Parameters - security index](#) 136
[Preconditions](#) 11
[Prerequisites](#) 11
[Product behavior](#) 137

Q

Queue
[alias](#) 25
[name - hash string calculation](#) 15

R

References
[informative](#) 10
[normative](#) 10
[Relationship to other protocols](#) 11

S

[Schema elements - directory service](#) 25
Security
[implementer considerations](#) 134
[parameter index](#) 136
[Sequencing rules](#) 30
[Standards assignments](#) 11

T

[Timer events](#) 30
[Timers](#) 29
[Tracking changes](#) 140
[Transport](#) 12

U

[Unused active directory properties](#) 15

V

[Vendor-extensible fields](#) 11
[Versioning](#) 11